AN INVERSE PROBLEM FOR NETWORKS

by

MICHAEL J. PARKER

S.B., Electrical Engineering, Massachusetts Institute of Technology (1988)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1990

Signature of Author .....
Department of Electrical Engineering and Computer Science
11 May 1990

Certified by .....
Dr. Gilbert Strang
Professor, Department of Mathematics
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chair, Department Committee on Graduate Students

AN INVERSE PROBLEM FOR NETWORKS

by

MICHAEL J. PARKER

Submitted to the Department of Electrical Engineering and Computer Science
on 11 May 1990, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering

Abstract

This thesis deals with an inverse problem for networks. We focus specifically on electrical networks – from measured responses to current or voltage sources, we try to find the conductances in a network. The topology (connectivity) of the network is assumed to be known.

In Chapter 1, basic terminology used in the inverse problem is introduced. We discuss the matrices involved in the inverse problem and present nonlinear equations which can be used to solve for the conductances. Requirements for existence and uniqueness of solutions to the inverse problem are discussed. It is shown that certain conditions for existence which can be found from graph theory can also be found from physical laws.

The subject of Chapter 2 is a numerical technique – Newton's method – which we use to solve the nonlinear equations for the conductances. After a brief overview of Newton's method, we discuss computer programs which have been written to solve the inverse problem for specific circuits. One of the programs uses an algorithm which applies Newton's method at each iteration, and we explain and discuss the algorithm.

In Chapter 3, we examine two matrices, here called P and R, which are essential to the inverse problem. The entries of P and R correspond to certain circuit measurements, and we present formulas for the entries of P and R in terms of the conductances. We show how Kirchhoff's Voltage Law determines the fundamental subspaces of P and R, and how those subspaces influence the inverse problem. Specifically, we find that only certain sets of measurements (corresponding to entries of P and R) contain enough information to solve the inverse problem. If Newton's method is attempted using a set of measurements which contain inadequate information, then the Jacobian matrix is singular and the method fails. An algorithm is presented to determine which sets of measurements can be used to solve the inverse problem.

Thesis Supervisor: Dr. Gilbert Strang
Title: Professor, Department of Mathematics

Acknowledgements

I wish to thank the following people for their help and guidance:

Dr. Gilbert Strang, my thesis supervisor, for his insight, guidance, and considerable kindness.

My Mother and Father, for their love and support throughout my years at MIT.

Barbara, for helping me keep a sense of humor about the whole thing.

Michael J. Parker
Cambridge, Massachusetts
May 1990

Contents

List of Figures

Introduction

In an "inverse" problem, an unknown system is probed with known sources, and responses are measured. From limited knowledge of the system and knowledge of the sources and responses, the goal is to reconstruct the system. The inverse problem is important in many areas. Geologists probe the earth by setting off explosions and measuring the underground shock waves which result. From these measurements, they try to determine the composition of the earth. In medical imaging, the human body is tested with X-rays or NMR or perhaps electrical currents, and an inverse problem is solved to form an image of what is inside the body. Our definition of the inverse problem is broad, and many more examples can be found.

In this thesis, we deal with the inverse problem for electrical networks. A network is probed with current or voltage sources, and from measured responses we try to find the conductances in the network. The connectivity of the network is assumed to be known, and is recorded in the connectivity matrix A. The edge conductances form a diagonal matrix C. The equilibrium equation underlying the problem is $A^TCA\vec{x} = \vec{f}$, where $\vec{f}$ is a vector of source terms and $\vec{x}$ is a vector of node voltages. In the "direct" problem, the conductances and sources are known, and we compute outputs (voltages or currents). In the inverse problem, only certain $\vec{x}$'s are known for certain $\vec{f}$'s and the idea is to reconstruct C.

In "Inverse Problems and Derivatives of Determinants"[7], Strang writes, "What is attractive is the appearance of a <u>convex potential function</u>, whose gradient is zero at the solution.... The logarithm of the determinant of A^TCA is the potential function for the inverse problem." The potential function, known to Kirchhoff, is essential in answering questions about existence and uniqueness of solutions to the inverse problem. One of the main questions dealt with in this thesis is which source/measurement combinations are sufficient to recover the conductances. The results may be important for realistic problems where certain edges of the network are inaccessible.

Chapter 1

1.1 Overview

The goal of Chapter 1 is to introduce terminology used in dealing with the inverse problem, and to present some of the theory behind the problem. Much of this chapter is adapted from a paper by Strang ([7]).

1.2 Terminology

We start with an example to introduce the matrices and vectors involved in the direct and inverse problems. In the inverse problem, it is desired to find conductance values in a network from a limited set of measured responses to known sources. In the direct problem, the conductances are known and it is desired to find the responses to known sources. The example we consider in shown in Figure 1-1. The four nodes (vertices) are labeled x_1, x_2, x_3, x_4, where x_i represents the potential (voltage) at node i. The six edges are labeled $y_1, y_2, y_3, y_4, y_5, y_6$, with y_i representing the current across edge i and the arrow representing the direction of flow.

The connectivity of the circuit is represented in the A_0 matrix. For the example, the connectivity matrix has the following form:

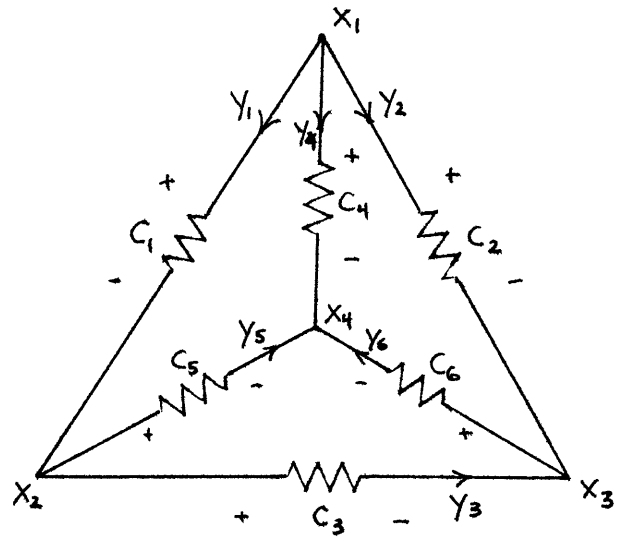


Figure 1-1: Example Circuit

$$A_0 = \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix}.$$

Each row corresponds to an edge and each column to a node. Edge 1 leaves node 1 and enters node 2 so there is a -1 in row 1, column 1 and a $+1$ in row 1, column 2 of A_0 . The other entries in row 1 are zero. The rest of A_0 is constructed the same way: $a_{ij}=$ -1 if edge i leaves node j, $+1$ if edge i enters node j, and 0 otherwise. A_0 is called the incidence matrix, or connectivity matrix or topology matrix for the network.

Multiplying the incidence matrix A_0 by the vector of potentials $\vec{x} = (x_1, x_2, x_3, x_4)$ gives a vector $\vec{v} = A_0\vec{x}$ containing potential differences. For Figure 1-1, $\vec{v} = (x_2 - x_1, x_3 - x_1, x_3 - x_2, x_4 - x_1, x_4 - x_2, x_4 - x_3)$. In subsequent calculations, it is only the potential differences which matter. We can therefore fix one of the node voltages, say $x_4 = 0$, to provide a reference against which other node voltages can be measured. This is called

"grounding" a node. The column of A_0 corresponding to the grounded node may be removed. In this case, after fixing $x_4 = 0$ we remove column 4 of A_0 leaving the new matrix

$$A = \begin{bmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}.$$

We can do this because the column of A_0 corresponding to the grounded node always gets multiplied by a zero voltage when the product $A_0\vec{x}$ is formed. The A_0 matrix has columns that add to the zero vector, but the truncated A matrix has independent columns. We then also leave $x_4 = 0$ out of the $\vec{x}$ vector so the dimensions of A and $\vec{x}$ agree.

The currents $y_1, y_2, ..., y_6$ are determined by the potential differences and the physical properties of the edges. The relevant physical property in Figure 1-1 is the conductance along each edge. These conductances are put into a diagonal matrix:

$$C = \begin{bmatrix} c_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & c_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_6 \end{bmatrix}.$$

In certain cases with coupling between edges (for example if the circuit contained mutual inductances) C might contain off-diagonal elements. The relation between the currents and the potential differences can be written in vector form $\vec{y} = -CA\vec{x}$. This is Ohm's Law. The minus sign indicates flow from higher potentials to lower potentials.

Vectors $\vec{b}$ and $\vec{f}$ are introduced to represent voltage and current sources respectively. Each component of $\vec{b}$ corresponds to an edge of the circuit. For the circuit in Figure 1-2,

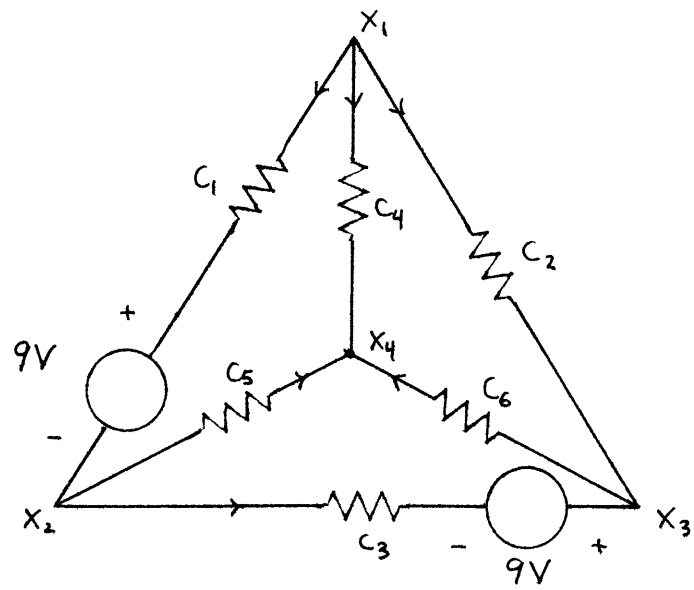


Figure 1-2: Circuit with voltage sources

$\vec{b} = (-9, 0, 9, 0, 0, 0)$. The component $b_1 = -9$ because the drop in voltage on edge 1 goes in the same direction as the current arrow. This convention keeps the following matrix equations consistent. The potential difference vector $\vec{v}$ can be related to the voltage sources $\vec{b}$ and node voltages $\vec{x}$ via the equation $\vec{v} = \vec{b} - A\vec{x}$. For Figure 1-2 (with node 4 grounded so that $x_4 = 0$) this equation gives

$$\vec{v} = \vec{b} - A\vec{x} = \begin{bmatrix} -9 \\ 0 \\ 9 \\ 0 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

This is Kirchhoff's Voltage Law (KVL) in vector form. If the voltage source were in *parallel* with edge i, that would fix the potential difference across that edge.

In the current source vector $\vec{f}$, each component corresponds to a node of the circuit. In general if current sources are injecting m units of current *into* node i then $f_i = -m$. If the current sources draw m units *out of* node i then $f_i = +$m. In Figure 1-3, $\vec{f} = (-3, 1, 2, 0)$.

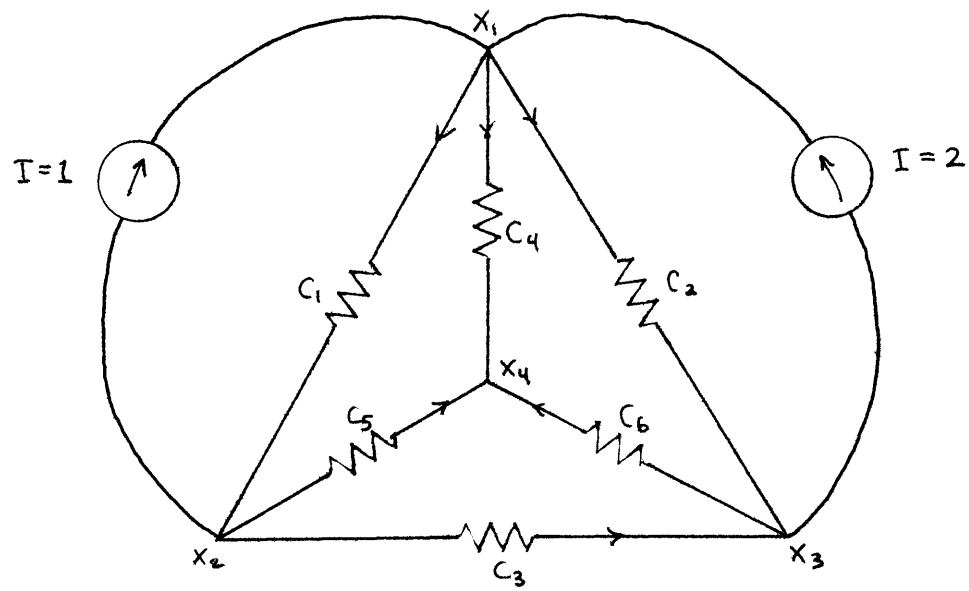


Figure 1-3: Circuit with current sources

There is a current source of magnitude 1 in parallel with edge 1 directed from node 2 to node 1. This gives $f_2 = 1$. There is a current source of strength 2 directed from node 3 to node 1, giving $f_3 = 2$. Together the two sources make $f_1 = -3$. The current sources can be related to the edge currents $\vec{y}$ by the equation $A^T\vec{y} = \vec{f}$. This is Kirchhoff's Current Law (KCL) in vector form. If a current source were placed in *series* with edge i, that would fix the flow y_i across that edge.

The basic equations are $\vec{v} = \vec{b} - A\vec{x}$ (KVL), $\vec{y} = C\vec{v}$ (Ohm's Law), and $A^T\vec{y} = \vec{f}$ (KCL). The relations between these vectors and matrices are well illustrated in Figure 1-4 ([6],p.91). Potentials $\vec{x} \Rightarrow$ potential differences $\vec{v} = \vec{b} - A\vec{x} \Rightarrow$ currents $\vec{y} = C\vec{v} \Rightarrow$ equilibrium equation $A^T\vec{y} = \vec{f}$. If there are no voltage sources, then $\vec{b} = (0,0,0,0,0,0)$ and we have that $\vec{f} = -A^TCA\vec{x}$.

The sources we primarily deal with in this paper are unit current sources in parallel with various edges. For such sources there is a convenient way to construct the $\vec{f}$ vector. Suppose, as in Figure 1-5, that there is a single unit current source in parallel with edge 1. We can then write $\vec{f}$ as $A^T\vec{e_1}$ where $\vec{e_1} = (1,0,0,0,0,0)$ is a unit vector with a 1 in the entry corresponding to the edge with the current source. Kirchhoff's Current Law

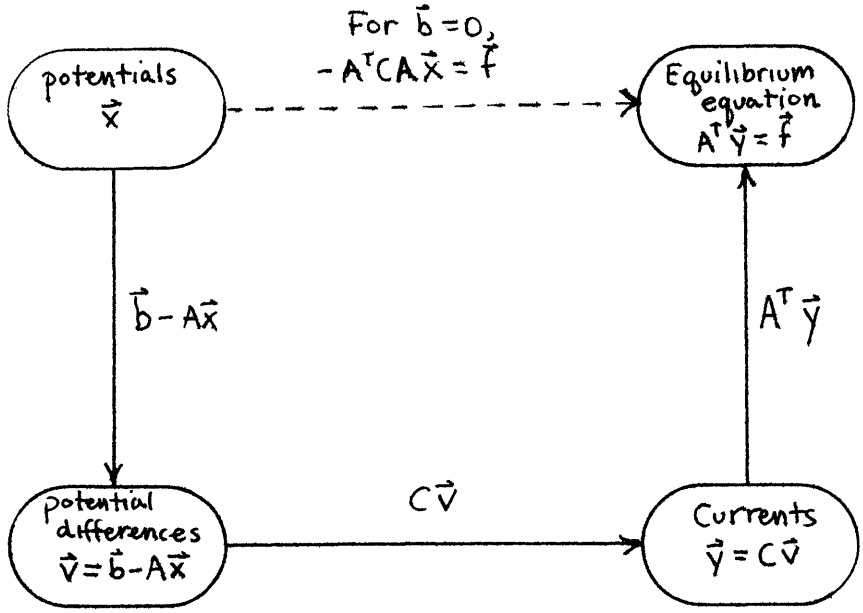


Figure 1-4: The overall framework

becomes $A^T\vec{y_j} = A^T\vec{e_j} = \vec{f}$, where $\vec{y_j}$ is the set of currents resulting from the unit current source in parallel with edge j (and no voltage sources). The vector $\vec{x_j}$ represents the potential differences resulting from such a source.

The above equations allow us to solve the direct problem. For unit current sources across edges j = 1, 2,..., m (each source separately) as above, we have $\vec{x_j} = -(A^TCA)^{-1}A^T\vec{e_j}$, $A\vec{x_j} = -A(A^TCA)^{-1}A^T\vec{e_j}$ where $A\vec{x_j}$ is a vector $\vec{v}$ of potential differences, and $\vec{y_j} = -CA\vec{x_j} = CA(A^TCA)^{-1}A^T\vec{e_j}$ where $\vec{y_j}$ is the vector of edge currents resulting from current source j. When C and A are known, one can solve for the potential differences $A\vec{x}$ and the currents $\vec{y}$.

1.3 The Inverse Problem

When C is not known, we have the inverse problem: the connectivity matrix A is known and outputs $\vec{x}$ and $\vec{y}$ are known for certain $\vec{f}$'s. From this information one tries to recover C. If we let $R = A(A^TCA)^{-1}A^T$ and $P = CA(A^TCA)^{-1}A^T$ then the equations from the previous paragraph can be written as $A\vec{x_j} = -R\vec{e_j}$ and $\vec{y_j} = P\vec{e_j}$.

From careful inspection of these last two equations, we find that R_{ij} (the i,j entry of

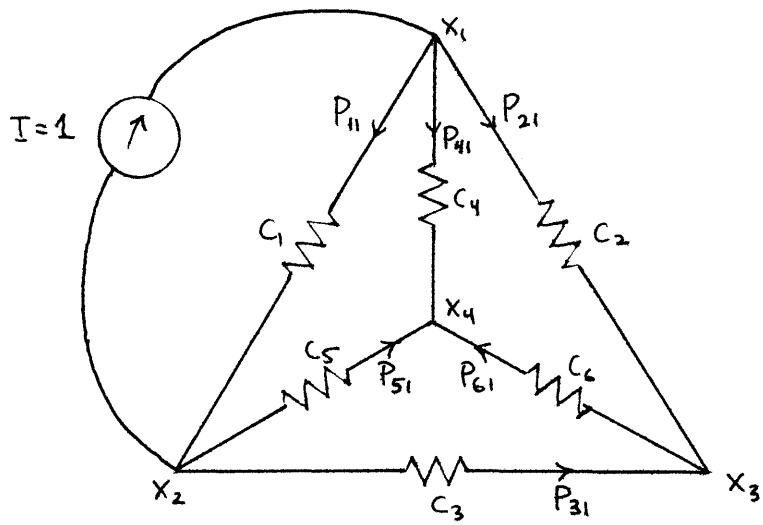


Figure 1-5: Single current source

the matrix R) can be thought of as the potential difference across edge i due to a unit current source in parallel with edge j. (The units of R_{ij} are resistance.) Similarly, P_{ij} is the current flowing through edge i due to a unit current source in parallel with edge j. The outputs we measure in the inverse problem correspond to certain entries of the R or P matrices. We would like to reconstruct C from knowledge of A and certain entries of R or P.

Two common types of measurements taken are referred to as the voltage probe and the current probe. A voltage probe measures the potential difference across edge i due to a unit current source in parallel with that same edge. This measurement corresponds to R_{ii}, a diagonal entry of R. A current probe measures the current flowing through edge i due to a unit current source in parallel with the same edge, and this measurement corresponds to P_{ii}, a diagonal entry of P. It is also called a voltage probe when one imposes a voltage and measures the resulting current – this gives the same information as the type of voltage probe mentioned above. Other types of probes involve measuring voltage across edge i due to a unit current source in parallel with edge j (this corresponds to R_{ij}), and measuring current through edge i due to a unit current source across edge j

(this corresponds to P_{ij}).

Let us apply current and voltage probes (separately) to the circuit in Figure 1-1. If a unit current source is placed across edge j and the resultant current flowing through edge j is measured, this measurement corresponds to P_{jj}. If all conductances in Figure 1-1 are equal to the constant c, then $P_{jj} = \frac{1}{2}$ for j = 1,2,...,6. A diagonal entry R_{jj} is equivalent to the system resistance or Thevenin resistance between the nodes of edge j. If all conductances equal c in Figure 1-1 then $R_{jj} = \frac{1}{2c}$ for j = 1,2,...,6. By inspection of the definitions of P and R, it is evident that P = CR. It follows that the diagonal entries of P and R are related by the equation $P_{jj} = c_j R_{jj}$ (for diagonal C).

P is a projection matrix (it satisfies $P^2 = P$). For projection matrices, the trace gives the number of linearly independent columns. For the circuit of Figure 1-1, $\Sigma_{i=1}^{6} P_{ii} = 3$ and P has only three independent columns (and three independent rows). Of the six diagonal entries, only five can be considered independent because of the constraint that the diagonal elements of P add to 3. This means that if one is trying to reconstruct C from the diagonal entries of P, the best one can do is to solve within a scale factor. If all conductances are multiplied by the same constant, the entries of P do not change.

Two questions are posed ([7],p.12) which will be answered shortly:

1. What range of values of P_{jj} j=1,2,...m (m-edge circuit) result from positive conductances?

2. Are the conductances uniquely determined (or determined to within a scale factor) from the P_{jj} (j = 1,2,...,m)?

We start by looking in particular at the voltage and current probe measurements corresponding to the diagonal elements of R and P because the equations for these entries have a simple form.

For the circuit of Figure 1-1,

$$P_{11} = \frac{c_1c_2c_4 + c_1c_2c_5 + c_1c_2c_6 + \textit{five other terms}}{c_1c_2c_4 + c_1c_2c_5 + c_1c_2c_6 + \textit{thirteen other terms}}. \tag{1.1}$$

Each product in the denominator corresponds to one of the sixteen spanning trees of the network. A spanning tree refers to a set of edges which *touch all nodes* but *form no loops.* For a circuit with n nodes there are $n-1$ edges in each spanning tree and n^{n-2} spanning trees (if the graph of the circuit is complete). The denominator of (1.1) is the sum of the conductance products from the sixteen spanning trees in the example network. This denominator, which will be referred to as Δ, is central to the inverse problem. Delta is the determinant of A^TCA, and appears in the equations because $(A^TCA)^{-1}$ is one of the matrices in $P = CA(A^TCA)^{-1}A^T$.

The numerator of (1.1) is the sum of the spanning trees which contain c_1 (the conductance on edge 1). In general, the numerator of P_{jj} is the sum of the spanning trees which include edge j. The denominator of P_{jj} is the same for all j: $\Delta = det(A^TCA)$. The equations for the P_{jj} can be conveniently represented as

$$P_{jj} = c_j \frac{\partial \log \Delta}{\partial c_j}. \tag{1.2}$$

From the relationship $P_{jj} = c_j R_{jj}$, we have that

$$R_{jj} = \frac{\partial \log \Delta}{\partial c_j}. \tag{1.3}$$

The diagonal of R therefore contains the gradient of $\log \Delta$ with respect to the c_j. The function $\log \Delta$ has been called the "impedance potential" ([1]) and its properties determine the answers to the existence and uniqueness questions posed above.

The diagonal entries of P contain the factors c_j multiplying the partial derivatives of $\log \Delta$. However, if we make a change of variables $c_j = e^{z_j}$ then the equations for the diagonal entries of P can be written

$$P_{jj} = \frac{\partial \log \Delta(z)}{\partial z_j}, \quad j = 1, 2, ..., m. \tag{1.4}$$

Substituting e^{z_j} for c_j yields $\Delta(z) = e^{z_1+z_2+z_4} + e^{z_1+z_2+z_5} + ... = \Sigma_T e^{\vec{t}\cdot\vec{z}}$, where T is the set

of all sixteen tree vectors $\vec{t}$, and $\vec{z} = (z_1, z_2, z_3, z_4, z_5, z_6)$. Each particular tree vector has the value +1 in the entries correponding to the edges in the spanning tree it represents. For example, the spanning tree with edges 1,2,4 would have the tree vector (1,1,0,1,0,0).

After the change to the z's, the diagonal of P contains the gradient of $\log \Delta$ with respect to the z_j. One advantage of the change to z's is that $c_j = e^{z_j}$ allows only positive conductances for real z's. Another advantage is that the impedance potential function $\log \Delta$ is *convex* in the z's.

The convexity of the impedance potential $\log \Delta(z)$ is proved ([7],p.15-16) by computing the first and second derivatives[1]:

$$\text{gradient}(\log \Delta(z)) = \frac{\Sigma_T \vec{t} e^{\vec{t}\cdot\vec{z}}}{\Sigma_T e^{\vec{t}\cdot\vec{z}}}$$

$$\text{Hessian}(\log \Delta(z)) = \frac{(\Sigma_T e^{\vec{t}\cdot\vec{z}})(\Sigma_T \vec{t}\vec{t}' e^{\vec{t}\cdot\vec{z}}) - (\Sigma_T \vec{t} e^{\vec{t}\cdot\vec{z}})(\Sigma_T \vec{t}' e^{\vec{t}\cdot\vec{z}})}{(\Sigma_T e^{\vec{t}\cdot\vec{z}})^2}.$$

The Schwarz inequality gives

$$(\Sigma_T e^{\vec{t}\cdot\vec{z}})(\Sigma_T (\vec{x}\cdot\vec{t})^2 e^{\vec{t}\cdot\vec{z}}) \geq (\Sigma_T (\vec{x}\cdot\vec{t}) e^{\vec{t}\cdot\vec{z}})^2$$

and the Hessian matrix is positive semidefinite ($\vec{x}' H \vec{x} \geq 0$ for any $\vec{x}$). It follows that $\log \Delta(z)$ is strictly convex except in the direction (1,1,1,1,1,1), where the Schwarz inequality becomes an equality. Multiplying all the conductances by a positive scale factor is equivalent to adding a multiple of (1,1,1,1,1,1) to $\vec{z}$, and the change has no effect on the elements of P. If the constraint $\Sigma z_j = 0$ is placed on the z_j, then the solution to the inverse problem $d_j = \frac{\partial}{\partial z_j} \log \Delta(z)$ is unique ([7],p.16). Thus the answer to question 2 is that except for a multiplicative scale factor in the c's (or an additive factor in the z's) the solution to $c_j \frac{\partial}{\partial c_j} \log \Delta = \frac{\partial}{\partial z_j} \log \Delta = d_j$ (j = 1,2,...,6) is unique.

There is still the question of which sets of currents come from positive conductances. From Strang ("Inverse Problems and Derivatives of Determinants",p.16), The-

[1] We use the symbol ' rather than ^{T} here to denote transposition

orem 3 states that $d_j = \frac{\partial}{\partial z_j} \log \Delta(z)$ (j = 1,2,...,m) "has a solution if and only if $\vec{d} = (d_1, d_2, d_3, d_4, d_5, d_6)$ is interior to the convex hull, on the hyperplane othogonal to (1,1,1,1,1,1), of the set T of tree vectors." The proof ([7],p.17) will not be restated here. The convex hull of T is described by the inequalities $\Sigma_{j=1}^{6} d_j = 3$, $0 < d_j < 1$, and the loop inequalities from the four loops (123, 145, 246, 356) in the circuit:

$$d_1 + d_2 + d_3 \leq 2, d_1 + d_4 + d_5 \leq 2, d_2 + d_4 + d_6 \leq 2, d_3 + d_5 + d_6 \leq 2.$$

A set of currents $d_1, d_2, ..., d_6$ which satisfy the above requirements is the result of a nonnegative set of conductances, and the inverse problem has a solution.

The four loop inequalities can be found from circuit theory. Kirchhoff's Voltage Law (KVL), which states that the voltage drop around any loop is zero, leads to the equalities:

$$P_{11} = P_{12} - P_{13}, P_{11} = P_{14} - P_{15}, P_{12} = P_{14} - P_{16}, P_{15} = P_{13} + P_{16}.$$

(For an explanation of how these equations are obtained, see Chapter 3. The signs result from the edge directions chosen in Figure 1-1.) We have that

$$d_1 + d_2 + d_3 = P_{11} + P_{22} + P_{33} = (P_{12} - P_{13}) + P_{22} + P_{33} \leq 2$$

because $P_{12} + P_{22} \leq 1$ and $-P_{13} + P_{33} \leq 1$. These last two inequalities can be found by inspection of the circuit. With a unit current source across edge 2, as in Figure 1-6, the current splits into several parts at node 1 – fractions of the unit current go through edges 1,2,4. By Kirchhoff's Current Law (KCL) , which states that current flowing into a node equals current flowing out of that node, we have that $P_{12} + P_{22} + P_{42} = 1$. Therefore $P_{12} + P_{22} \leq 1$, with equality only if there is no current flowing through edge 4. By KCL we also have that $-P_{13} + P_{33} + P_{53} = 1$ so that $-P_{13} + P_{33} \leq 1$. Thus, the first loop inequality is shown to be true. The other loop inequalities can be proven using KVL and KCL in the same way ($d_1 + d_4 + d_5 = P_{11} + P_{44} + P_{55} = (P_{14} - P_{15}) + P_{44} + P_{55} \leq 2$

because $P_{14} + P_{44} \leq 1$ and $-P_{15} + P_{55} \leq 1$, etc.).

The convex hull requirements can be found by graph theory or by KVL and KCL. The result is that a set of currents $d_1, d_2, ..., d_6$ which satisfy $0 \leq d_j \leq 1$, $\Sigma_j d_j = 3$, and the loop inequalities obeys physical principles (KVL and KCL) and is therefore within the range of possible results. These convex hull requirements can be found via graph theory for any circuit, and probably also using KVL and KCL. It is a reasonable conjecture that only currents which obey KVL and KCL are possible in a circuit with positive conductances.

Questions 1 and 2 about uniqueness and existence have been answered. The possible sets of current probe data are those which satisfy the convex hull requirements, and for those current probe measurements, the solution to the inverse problem is unique to within a scale factor. When we impose the constraint $\prod c_j = 1$ (or $\Sigma z_j = 0$), there is uniqueness in the c's and z's. Chapter 2 deals with using Newton's method to solve the inverse problem.

Chapter 2

2.1 Overview

Chapter 2 deals with a numerical method of solving for the conductances from knowledge of the diagonal entries of the P or R matrices. A unit current source is placed in parallel with each edge of the circuit in turn. The current flowing through the edge with the source is measured (current probe $\rightsquigarrow$ diagonal element of P), or the voltage across the edge with the source is measured (voltage probe $\rightsquigarrow$ diagonal element of R). From these measurements, we would like to find the conductances in the network. If the connectivity of the network is known, then the formulas for the diagonal elements of R and P are known. From Chapter 1, we have that $R_{jj} = \frac{\partial}{\partial c_j} \log \Delta$ and $P_{jj} = c_j \frac{\partial}{\partial c_j} \log \Delta$, where $\Delta = det(A^T C A)$. It is these equations which are used to recover the conductances. Although the computer programs are written for current probe data, the method for solving from voltage probe data is essentially the same.

2.2 Newton's Method

The following summary of Newton's method uses the notation and approach found in Introduction to Applied Mathematics ([6], pp.373-4). The equations for the diagonal elements of P and R are nonlinear in the c's. The technique we use to solve these nonlinear equations – Newton's method – starts with an initial guess and improves upon it by using

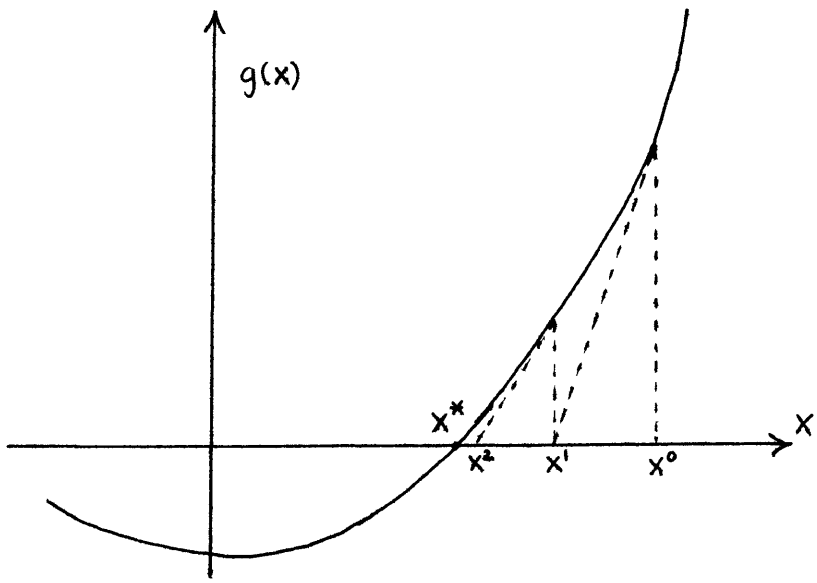


Figure 2-1: Newton's method (diagram from [6],p.374)

linear approximations to the nonlinear equations. To construct the approximations, the function and its derivative are needed. For example, to solve the one-dimensional equation $g(x) = 0$, Newton's method starts with an initial guess x^1 (superscript, not power). The tangent line approximation to $g(x)$ at x^1 is $g(x^2) \approx g(x^1) + g'(x^1)(x^2 - x^1)$ (see Figure 2-1). To find out where that line crosses the horizontal axis (which will give the next guess), set $g(x^2) \approx 0$ so that $g'(x^1)(x^2 - x^1) = -g(x^1)$. The new guess x^2 is then improved to x^3 (unless $g(x)$ is linear and $g(x^2) = 0$) and the process repeats until a reasonable approximation to a solution is obtained.

The same technique applies to higher dimensional equations. If there are n functions $g_1, g_2, ..., g_n$ and n unknowns $c_1, c_2, ..., c_n$, the surfaces described by $g_1 = 0, g_2 = 0, ..., g_n = 0$ intersect in a curve which intersects the horizontal at $c^* = (c_1^*, c_2^*, ..., c_n^*)$. An initial guess $\vec{c}^1$ can be improved by constructing the tangent plane to each surface at $\vec{c}^1$. These tangent planes meet in a line which intersects the horizontal at $\vec{c}^2$, which is the next guess. The equations for the tangent planes can be represented in a matrix called the

Jacobian:

$$J^k = \begin{bmatrix} \frac{\partial g_1}{\partial c_1} & \frac{\partial g_1}{\partial c_2} & \cdots\cdots & \frac{\partial g_1}{\partial c_n} \\ \frac{\partial g_2}{\partial c_1} & \frac{\partial g_2}{\partial c_2} & \cdots\cdots & \frac{\partial g_2}{\partial c_n} \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial g_n}{\partial c_1} & \frac{\partial g_n}{\partial c_2} & \cdots\cdots & \frac{\partial g_n}{\partial c_n} \end{bmatrix}_{\vec{c}^k} .$$

The notation shown refers to the Jacobian evaluated at the guess $\vec{c}^k$. The equation for the next guess is then

$$J^k(\vec{c}^{k+1} - \vec{c}^k) = -\vec{g}^k = -(g_1(c_1, ..., c_n), g_2(c_1, ..., c_n), ..., g_n(c_1, ..., c_n)). \tag{2.1}$$

Newton's method may not converge to a solution. Convergence depends highly on the initial guess. If the initial guess is in the "basin of attraction" or "region of convergence" of a solution, then Newton's method will find that solution. If there is more than one solution, each will have its basin of attraction and it may be difficult to find all the solutions. If the initial guess is not in the basin of attraction of any solution, then Newton's method diverges.

We first set up Newton's method for the three resistor ring shown in Figure 2-2. Divergence of initial guesses is shown to be a problem. The equations for the current probe data are

$$P_{11} = \frac{c_1c_2 + c_1c_3}{c_1c_2 + c_1c_3 + c_2c_3} = d_1$$

$$P_{22} = \frac{c_1c_2 + c_2c_3}{c_1c_2 + c_1c_3 + c_2c_3} = d_2$$

$$P_{33} = \frac{c_1c_3 + c_2c_3}{c_1c_2 + c_1c_3 + c_2c_3} = d_3$$

where d_1, d_2, d_3 are the measured values of P_{11}, P_{22}, P_{33}. The three nonlinear equations are

$$g_1(c_1, c_2, c_3) = \frac{c_1c_2 + c_1c_3}{c_1c_2 + c_1c_3 + c_2c_3} - d_1 = 0$$

$$g_2(c_1, c_2, c_3) = \frac{c_1c_2 + c_2c_3}{c_1c_2 + c_1c_3 + c_2c_3} - d_2 = 0$$

$$g_3(c_1, c_2, c_3) = \frac{c_1c_3 + c_2c_3}{c_1c_2 + c_1c_3 + c_2c_3} - d_3 = 0.$$

The general form of the Jacobian contains the partial derivatives of these functions as displayed previously. The Jacobian formed from the above three equations is singular, and therefore cannot be used in Newton's method. The diagonal elements of P sum to two, and taking the derivative of the equation $P_{11} + P_{22} + P_{33} = 2$ with respect to any of the c's gives

$$\frac{\partial P_{11}}{\partial c_i} + \frac{\partial P_{22}}{\partial c_i} + \frac{\partial P_{33}}{\partial c_i} = 0.$$

Thus every column of the Jacobian sums to zero, and the rank is less than 3. It was observed in Chapter 1 that the (diagonal) elements of P do not contain enough information to uniquely determine the conductances. Multiplying all conductances by a constant leaves the P matrix unchanged. In order for Newton's method to give a unique answer, the constraint equation

$$g(c_1, c_2, ..., c_m) = \prod_{i=1}^{m} c_i - 1 = 0$$

is used instead of one of the other nonlinear equations $(g_1, g_2, ..., g_m)$=0 (where m is the number of edges in the circuit). In the case of the three-resistor ring, the equation $g(c_1, c_2, c_3) = c_1c_2c_3 - 1 = 0$ would replace any one of the three original equations.

2.3 The Computer Programs

The computer program NEWTON3 listed in Appendix B solves the inverse problem for the three-resistor ring. It takes as inputs the values of P_{11}, P_{22}, P_{33} and initial guesses for the conductances. The program then uses the formula (2.1) to try and improve the initial guess $\vec{c}^1$ until the values of g_1, g_2, g_3 are all close to zero (the program uses tolerance .001). If Newton's method converges, then the program finds a solution to the inverse problem, and outputs the values of the conductances.

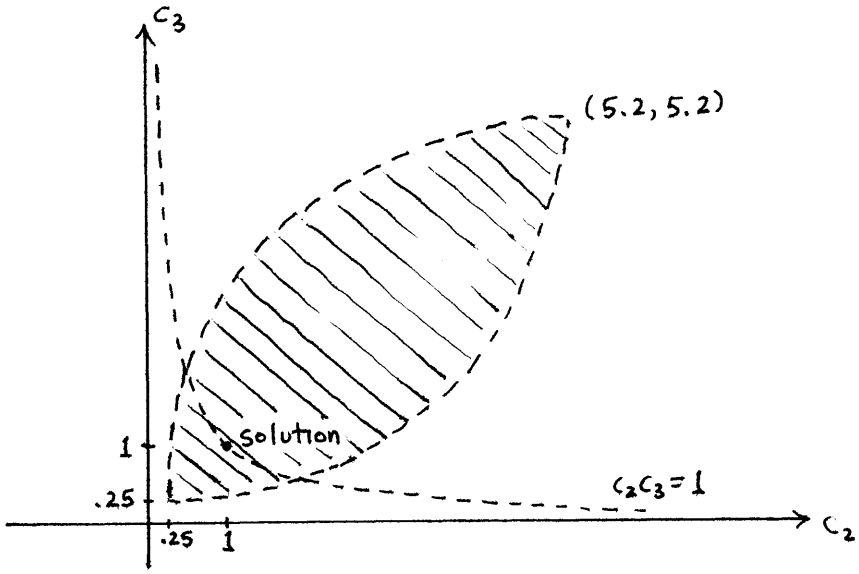


Figure 2-2: Region of convergence in plane $c_1 = 1$.

Let us now examine the region of convergence for Newton's method applied to the inverse problem for the three-resistor ring. The actual conductances are taken to be $(c_1, c_2, c_3) = (1, 1, 1)$. For many initial guesses, such as $(c_1, c_2, c_3) = (2, 6, 2)$ and $(c_1, c_2, c_3) = (1, 3, 5)$, Newton's method diverges. Since finding and describing regions of convergence even in three dimensions can be difficult, we restrict the search to the plane $c_1 = 1$. The constraint equation becomes $c_2c_3 = 1$ and we are looking for solutions along the line described by that equation (see Figure 2-2). The region of convergence in the plane $c_1 = 1$ is plotted experimentally and is shown in Figure 2-2. An initial guess outside the shaded regions causes Newton's method to diverge.

The computer program to solve the inverse problem for the six edge/four node circuit of Figure 1-1 is called NEWTON6A and is listed in Appendix B. Like the program for the three-resistor ring, NEWTON6A takes as input the values of the diagonal elements of P – the current probe data. However, the algorithm is improved. The program first supplies a more intelligent initial guess, before running Newton's method in the same fashion as NEWTON3. The current probe data is used as the initial guess: $(c_1, c_2, ..., c_6) = (P_{11}, P_{22}, ..., P_{66})$. The rationale is that the currents should be to some degree proportional

to the conductances on each edge. If the initial guess diverges, then a different approach, suggested to this author by Professor Alar Toomre of MIT, is tried.

The new approach starts with a problem which has a known solution – in the six edge/four node case we know that the currents $(P_{11}, P_{22}, ..., P_{66}) = (.5, .5, .5, .5, .5, .5)$ come from equal conductances $(c_1, c_2, ...c_6) = (1, 1, 1, 1, 1, 1)$. That problem is then varied slightly toward the problem to be solved. For example, the current fractions are changed to (.502,.498,.502,.498,.502,.498) if the ultimate goal is to solve the inverse problem for $(P_{11}, P_{22}, ..., P_{66}) = (.9, .1, .9, .1, .9, .1)$. Using the previous solution (in this case (1,1,1,1,1,1)) as an initial guess, Newton's method solves the intermediate problem. The current fractions are then varied again (continuing the example, they are changed to (.505,.495,.505,.495,.505,.495)), and the latest intermediate solution is used as an initial guess for Newton's method. By slowly changing the current fractions toward the data for which a solution is desired, and by using the latest intermediate solution as an initial guess, the program should eventually reach a solution for the actual current probe data.

Sometimes the intermediate current fractions are varied too far and Newton's method diverges for an intermediate problem. In that case, the program goes back and changes the currents by smaller amounts. If the intermediate problem still diverges, the program makes even smaller changes. If, after a certain number of tries, the intermediate problem still diverges, then the program acknowledges that it could not find the solution. The steps of NEWTON6A are illustrated by the diagram shown in Figure 2-3. If the initial guess had failed, the diagram shows the progression of the intermediate problems toward a final solution. The stepping process is comparatively slow, since it performs Newton's method many times. Running an intelligent initial guess through Newton's method is an attempt to help the program find a solution more quickly.

To test the performance of NEWTON6A, various sets of hypothetical current probe data are tried. Values near the edge of the convex hull of tree vectors (see Chapter 1) are tested, and the results are shown in Figure 2-4. The same method applies to the inverse problem for any network and any set of measurements, as long as the measurements

$(P_{11}, P_{22}, P_{33}, P_{44}, P_{55}, P_{66})$ $\qquad$ $(c_1, c_2, c_3, c_4, c_5, c_6)$

$$
\begin{tikzcd}
(.500, .500, .500, .500, .500, .500) \arrow[r, "\text{Newton's method}"] & (1.00, 1.00, 1.00, 1.00, 1.00, 1.00) \arrow[dl, "\text{initial guess}"'] \\
(.502, .498, .502, .498, .502, .498) \arrow[r] & (1.01, 0.99, 1.01, 0.99, 1.01, 0.99) \arrow[dl] \\
(.505, .495, .505, .495, .505, .495) \arrow[r] & (1.02, 0.98, 1.02, 0.98, 1.02, 0.98) \arrow[dl] \\
(.511, .489, .511, .489, .511, .489) \arrow[r] & (1.04, 0.96, 1.04, 0.96, 1.04, 0.96) \arrow[dl] \\
(.523, .477, .523, .477, .523, .477) \arrow[r] & (1.09, 0.92, 1.09, 0.92, 1.09, 0.92) \arrow[dl] \\
(.547, .453, .547, .453, .547, .453) \arrow[r] & (1.19, 0.84, 1.19, 0.84, 1.19, 0.84) \arrow[dl] \\
(.591, .409, .591, .409, .591, .409) \arrow[r] & (1.37, 0.73, 1.37, 0.73, 1.37, 0.73) \arrow[dl] \\
(.668, .332, .668, .332, .668, .332) \arrow[r] & (1.74, 0.57, 1.74, 0.57, 1.74, 0.57) \arrow[dl] \\
(.784, .216, .784, .216, .784, .216) \arrow[r] & (2.50, 0.40, 2.50, 0.40, 2.50, 0.40) \arrow[dl] \\
(.900, .100, .900, .100, .900, .100) \arrow[r] & (4.12, 0.24, 4.12, 0.24, 4.12, 0.24)
\end{tikzcd}
$$

Figure 2-3: Stepping toward a solution

$(P_{11}, P_{22}, P_{33}, P_{44}, P_{55}, P_{66})$	$(c_1, c_2, c_3, c_4, c_5, c_6)$
$(0.60, 0.60, 0.60, .040, 0.40, 0.40)$	$(1.73, 1.73, 1.73, 0.58, 0.58, 0.58)$
$(0.65, 0.65, 0.65, 0.35, 0.35, 0.35)$	$(3.60, 3.60, 3.60, 0.28, 0.28, 0.28)$
$(0.66, 0.66, 0.66, 0.34, 0.34, 0.34)$	$(5.74, 5.74, 5.74, 0.17, 0.17, 0.17)$
$(0.67, 0.67, 0.67, 0.33, 0.33, 0.33)$	solution not found (doesn't exist)
$(0.90, 0.90, 0.10, 0.90, 0.10, 0.10)$	$(4.12, 4.12, 0.24, 4.12, 0.24, 0.24)$
$(0.97, 0.97, 0.03, 0.97, 0.03, 0.03)$	$(7.98, 7.98, 0.13, 7.98, 0.13, 0.13)$
$(0.99, 0.99, 0.01, 0.99, 0.01, 0.01)$	$(14.0, 14.0, 0.07, 14.0, 0.07, 0.07)$
$(.995, .995, .005, .995, .005, .005)$	$(19.9, 19.9, 0.05, 19.9, 0.05, 0.05)$
$(.999, .999, .001, .999, .001, .001)$	solution not found

Figure 2-4: Results from NEWTON6A

contain enough information to solve the inverse problem. According to Strang ([7]), we can solve the inverse problem from m diagonal entries of R, or m-1 diagonal entries of P and a constraint equation. Off-diagonal entries from R or P cannot be chosen at random, however, as will be discussed in Chapter 3.

Chapter 3

3.1 Overview

We have so far only dealt with measurements on the diagonals of P and R. This chapter asks about the rest of the entries in P and R – how these entries are related to each other and which measurements are sufficient to solve the inverse problem using Newton's method. These questions are looked at in specific for the six edge/four node circuit of Figure 1-1 (which we henceforth just call circuit A for convenience), and some general conclusions are drawn from these findings.

3.2 Entries of P and R

The diagonal entries of P and R come from current and voltage probe measurements respectively. As previously stated, the formulas for these entries are $P_{jj} = c_j \frac{\partial \log \Delta}{\partial c_j}$ and $R_{jj} = \frac{\partial \log \Delta}{\partial c_j}$, where $\Delta = det(A^T C A)$. The change of variables $c_j = e^{z_j}$ made it clear that a solution to the inverse problem using the current probe data is unique to within a scale factor.

The off-diagonal entries of P and R come from cross-measurements – measurements across edges not corresponding to the unit current or voltage probes. For example, P_{21} is the current flowing through c_2 due to a unit current source in parallel with c_1. The

general formulas for the off-diagonals are

$$P_{ij} = c_i\sqrt{-\frac{\partial^2 \log \Delta}{\partial c_i \partial c_j}} \tag{3.1}$$

and

$$R_{ij} = \sqrt{-\frac{\partial^2 \log \Delta}{\partial c_i \partial c_j}}. \tag{3.2}$$

These equations were found by inspection, but the result is proven by Strang ([7], pp.6-7). For circuit A, the equations resulting from these formulas are rather simple – the part under the square root is always a perfect square. The typical form of equations for the entries of P can be seen by looking at the equations for column one of P:

$$P_{11} = \frac{c_1c_2c_4 + c_1c_2c_5 + c_1c_2c_6 + c_1c_3c_4 + c_1c_3c_5 + c_1c_3c_6 + c_1c_4c_6 + c_1c_5c_6}{\Delta}$$

$$P_{21} = \frac{c_2(c_3c_4 + c_3c_5 + c_3c_6 + c_5c_6)}{\Delta}$$

$$P_{31} = -\frac{c_3(c_2c_4 + c_2c_5 + c_2c_6 + c_4c_6)}{\Delta}$$

$$P_{41} = \frac{c_4(c_3c_5 + c_2c_5 + c_3c_6 + c_5c_6)}{\Delta}$$

$$P_{51} = -\frac{c_5(c_2c_4 + c_3c_4 + c_2c_6 + c_4c_6)}{\Delta}$$

$$P_{61} = -\frac{c_6(c_3c_4 - c_2c_5)}{\Delta}.$$

(The signs result from the edge directions chosen in Figure 1-1.) The entries of the R matrix have a similar form, but a factor c_i is removed from each entry: $R_{ij} = P_{ij}/c_i$.

The diagonal elements of P and R also obey the general formulas for the off-diagonals, as one would expect. The general formula (3.1) applied for a diagonal element gives $P_{ii} = c_i\sqrt{-\frac{\partial^2 \log \Delta}{\partial c_i{}^2}}$, and we show that $P_{ii} = c_i\frac{\partial \log \Delta}{\partial c_i}$ is really the same equation, i.e. that $\sqrt{-\frac{\partial^2 \log \Delta}{\partial c_i{}^2}} = \frac{\partial \log \Delta}{\partial c_i}$. Expanding both sides gives $\sqrt{\frac{(\frac{\partial \Delta}{\partial c_i})^2 - \Delta\frac{\partial^2 \Delta}{\partial c_i{}^2}}{\Delta^2}} = \frac{\frac{\partial \Delta}{\partial c_i}}{\Delta}$. This is true because $\frac{\partial^2 \Delta}{\partial c_i{}^2}$ equals zero (all c_i only appear to first order in Δ). The general P_{ij} and R_{ij} formulas

therefore encompass the diagonal elements.

We also observe that certain entries of the P and R matrices for the example circuit of Figure 1-1 have simpler equations than others. The elements P_{61}, P_{52}, P_{43}, P_{34}, P_{25}, P_{16} have simpler formulas than the other entries. Whereas the equations for all other entries involve all six conductances in their numerators, those six entries' numerators only involve five of the conductances and in a simple form. The reason for the simpler form is that these entries come from measurements on edges symmetrically opposite the current sources. Looking at circuit A, we see that c_6 is symmetrically opposite c_1, c_5 is opposite c_2, and c_4 is opposite c_3. The significance of this symmetry is that the current through (or voltage across) a resistor which is opposite a source is not sensitive to changes in certain resistors in the network. For example, in circuit A, the measurement corresponding to P_{61} is much less sensitive to changes in c_6 and c_1 than to changes in other resistors, although the actual sensitivity depends on the resistor values themselves. The issue of sensitivity to changes in certain resistors is discussed in Appendix A.

3.3 The Fundamental Subspaces of P and R

We now examine the question of which entries of P (or R) are sufficient to solve the inverse problem using Newton's method. In Chapter 2, we saw that the diagonal elements of P contain enough information to solve the inverse problem to within a scale factor, and from Chapter 1 it is known that P is a projection matrix with rank $n-1$ (for a connected network with n nodes). For circuit A, only three columns of P are independent – the others are linear combinations of the independent columns. The same is true for the rows of P, i.e. there are only three independent rows and the others are simply linear combinations of those three.

Which columns and rows of P and R are independent, i.e. form a basis for the space spanned by the columns of the matrix? To answer that question, we find which vectors lie in the nullspaces ($P\vec{x} = 0$, $R\vec{x} = 0$) and left nullspaces ($P^T\vec{y} = 0$, $R^T\vec{y} = 0$) of P and

R.

The Nullspace of P From the definition in Chapter 1, $P = CA(A^TCA)^{-1}A^T$ and it is evident that any vector satisfying $A^T\vec{x} = 0$ also satisfies $P\vec{x} = 0$. The vectors $\vec{x_1} = (1,-1,1,0,0,0)$, $\vec{x_2} = (1,0,0,-1,1,0)$, $\vec{x_3} = (0,1,0,-1,0,1)$, $\vec{x_4} = (0,0,1,0,-1,1)$ all satisfy $P\vec{x} = 0$, and any three $\vec{x}$'s form a basis for the nullspace of P. These nullvectors $\vec{x_1}, \vec{x_2}, \vec{x_3}, \vec{x_4}$ come from the *loops* of the circuit: $\vec{x_1}$ represents the loop formed by edges 1,2,3 in circuit A, $\vec{x_2}$ represents the loop formed by edges 1,4,5, $\vec{x_3}$ represents the loop formed by edges 2,4,6, and $\vec{x_4}$ represents the loop formed by edges 3,5,6. Each $\vec{x}$ vector corresponds to tracing around a loop in a certain direction. Each $+1$ in $\vec{x}$ corresponds to going in the same direction as the current arrow on that edge, and each -1 corresponds to going in the opposite direction. The vector $\vec{x_1}$, for example, is formed by tracing counterclockwise around the loop formed by edges 1,2,3. Tracing around the same loop in the clockwise direction would give an equally valid nullvector $(-1,1,-1,0,0,0)$ (just the opposite of the previous $\vec{x_1}$). From $P\vec{x} = 0$, we have the relations for circuit A: $P_{i1} - P_{i2} + P_{i3} = 0, P_{i1} - P_{i4} + P_{i5} = 0, P_{i2} - P_{i4} + P_{i6} = 0, P_{i3} - P_{i5} + P_{i6} = 0$ for i=1,2,...,6.

The Left Nullspace of P The left nullspace of P – the relationship between the rows – comes directly from Kirchhoff's Voltage Law. Kirchhoff's Voltage Law (KVL) states that the total drop in voltage around any closed loop of a circuit must be zero. By Ohm's Law, $\frac{P_{ij}}{c_i}$ is the voltage across c_i due to a unit current source in parallel with c_j. For the loop formed by edges 1,2,3 in our circuit, we can write KVL: $\frac{P_{1i}}{c_1} - \frac{P_{2i}}{c_2} + \frac{P_{3i}}{c_3} = 0$. It is arbitrarily chosen which end of each resistor is to be considered higher in potential, and that leads to the positive and negative signs in KVL. Writing KVL for the other loops gives $\frac{P_{1i}}{c_1} - \frac{P_{4i}}{c_4} + \frac{P_{5i}}{c_5} = 0$, $\frac{P_{2i}}{c_2} - \frac{P_{4i}}{c_4} + \frac{P_{6i}}{c_6} = 0$, $\frac{P_{3i}}{c_3} - \frac{P_{5i}}{c_5} + \frac{P_{6i}}{c_6} = 0$. These equations could also have been derived from the formulas at the end of the previous paragraph by using the relation $P_{ij} = \frac{c_i}{c_j}P_{ji}$. The left nullspace of P is thus spanned by any three of the following four vectors: $\vec{y_1} = (\frac{1}{c_1}, -\frac{1}{c_2}, \frac{1}{c_3}, 0, 0, 0), \vec{y_2} = (\frac{1}{c_1}, 0, 0, -\frac{1}{c_4}, \frac{1}{c_5}), \vec{y_3} =$

$(0, \frac{1}{c_2}, 0, -\frac{1}{c_4}, 0, \frac{1}{c_6}), \vec{y_4} = (0, 0, \frac{1}{c_3}, 0, -\frac{1}{c_5}, \frac{1}{c_6})$. These vectors can also be found by testing $A^T C\vec{y} = 0$, because $P^T = A(A^T CA)^{-T} A^T C$ ends with the product $A^T C$. We have thus found that the nullspace and left nullspace of P are determined by the *loops* in the circuit: any set of rows or columns corresponding to edges which form a loop are dependent, i.e. some linear combination of those rows or columns gives zero.

The Nullspace and Left Nullspace of R The relationships between the rows and columns of R also come from KVL. Using the equation $R_{ij} = \frac{P_{ij}}{c_i}$, the KVL equations in the previous paragraph can be rewritten as $R_{1i} - R_{2i} + R_{3i} = 0, R_{1i} - R_{4i} + R_{5i} = 0, R_{2i} - R_{4i} + R_{6i} = 0, R_{3i} - R_{5i} + R_{6i} = 0$. The left nullspace of R is precisely the same as the nullspace of P and is spanned by any three of the same four $\vec{x}$ vectors. The R matrix is symmetric ($R^T = R$) so any three of the $\vec{x}$ vectors also constitute a basis for the nullspace of R. This result also follows from looking at the definition $R = -A(A^T CA)^{-1} A^T$. Any vector satisfying $A^T\vec{x}$=0 satisfies both $R\vec{x}$=0 and $R^T\vec{x}$=0 and therefore the nullspace and left nullspace of R can be found from $A^T\vec{x}$=0. The conclusion for R is the same as for P: any set of rows or columns which correspond to the edges of a loop are dependent, i.e. some linear combination of those rows or columns gives zero.

3.4 The Jacobian and Newton's Method

To solve the inverse problem for circuit A, one tries to choose six independent entries from P (the minimum number to recover $c_1, c_2, ..., c_6$) and use Newton's method to solve for the conductances. For Newton's method to operate, it is necessary that the Jacobian matrix J be nonsingular. Actually, we already know that the conductances cannot be determined uniquely from the P matrix – if all conductances are multiplied by a scale factor, P remains unchanged. We now examine which combinations of entries from P lead to singular Jacobian matrices. The same question is looked at for Jacobians formed from entries of R.

Using Entries From a Row of P Suppose we choose three (or more) entries from a single row of P. If the entries correspond to edges which form a loop (for example, P_{11}, P_{12}, P_{13} correspond to edges 1,2,3) then J will be singular. Choosing the entries $P_{11}, P_{12}, P_{13}, P_{24}, P_{45}, P_{66}$ in the case of circuit A leads to a Jacobian of the form

$$J = \begin{bmatrix} \frac{\partial P_{11}}{\partial c_1} & \frac{\partial P_{11}}{\partial c_2} & \frac{\partial P_{11}}{\partial c_3} & \frac{\partial P_{11}}{\partial c_4} & \frac{\partial P_{11}}{\partial c_5} & \frac{\partial P_{11}}{\partial c_6} \\ \frac{\partial P_{12}}{\partial c_1} & \frac{\partial P_{12}}{\partial c_2} & \frac{\partial P_{12}}{\partial c_3} & \frac{\partial P_{12}}{\partial c_4} & \frac{\partial P_{12}}{\partial c_5} & \frac{\partial P_{12}}{\partial c_6} \\ \frac{\partial P_{13}}{\partial c_1} & \frac{\partial P_{13}}{\partial c_2} & \frac{\partial P_{13}}{\partial c_3} & \frac{\partial P_{13}}{\partial c_4} & \frac{\partial P_{13}}{\partial c_5} & \frac{\partial P_{13}}{\partial c_6} \\ \frac{\partial P_{24}}{\partial c_1} & \frac{\partial P_{24}}{\partial c_2} & \frac{\partial P_{24}}{\partial c_3} & \frac{\partial P_{24}}{\partial c_4} & \frac{\partial P_{24}}{\partial c_5} & \frac{\partial P_{24}}{\partial c_6} \\ \frac{\partial P_{45}}{\partial c_1} & \frac{\partial P_{45}}{\partial c_2} & \frac{\partial P_{45}}{\partial c_3} & \frac{\partial P_{45}}{\partial c_4} & \frac{\partial P_{45}}{\partial c_5} & \frac{\partial P_{45}}{\partial c_6} \\ \frac{\partial P_{66}}{\partial c_1} & \frac{\partial P_{66}}{\partial c_2} & \frac{\partial P_{66}}{\partial c_3} & \frac{\partial P_{66}}{\partial c_4} & \frac{\partial P_{66}}{\partial c_5} & \frac{\partial P_{66}}{\partial c_6} \end{bmatrix}.$$

Since $P_{i1} - P_{i2} + P_{i3} = 0$, it follows that $\frac{\partial P_{i1}}{\partial c_j} - \frac{\partial P_{i2}}{\partial c_j} + \frac{\partial P_{i3}}{\partial c_j} = 0$. In the same way, we can see that any row entries which are linearly dependent in P will still be dependent in J (but as members of a column).

Using Entries From a Column of P Suppose we choose three (or more) entries from a single column of P. The requirements for J to be nonsingular are still simple, but different than the requirements if entries are selected from a row as above. The condition is that if the three or more entries from a column of P correspond to a set of edges related by Kirchhoff's Current Law (KCL), then the resulting J will be singular. Kirchhoff's Current Law states that total current flow into any node equals total current flow out of that node. If P_{11}, P_{12}, P_{14} are among the entries chosen in the case of circuit A, then J is singular because edges 1,2,4 include all the edges coming from node 1 and the currents through those edges are linearly dependent by KCL. All entries in column i of P are due to a unit current source in parallel with edge i. Using KCL, one can find relationships between those entries. For example, in column 1 of the P matrix for circuit A, we have that $P_{11}+P_{21}+P_{41} = 1, P_{11}-P_{31}-P_{51} = 1, P_{21}+P_{31} = P_{61}, P_{41}+P_{51}+P_{61} = 0$. (The current source associated with column 1 is shown in Figure 1-5). For each column,

the relationships are different, though, because each column is the result of a different current course. In column 3, KCL dictates that $P_{13} + P_{23} + P_{43} = 0, -P_{13} + P_{33} + P_{53} = 1, P_{23} + P_{33} - P_{63} = 1, P_{41} + P_{51} + P_{61} = 0$. If we choose any three (or more) entries from a column of P which are directly related by KCL, the relationship will still hold in the Jacobian. For example, if P_{11}, P_{21}, P_{41} are chosen, then the derivative of $P_{11}+P_{21}+P_{41} = 1$ gives $\frac{\partial P_{11}}{\partial c_j} - \frac{\partial P_{21}}{\partial c_j} + \frac{\partial P_{41}}{\partial c_j} = 0$, showing how the column entries of J will be dependent.

It should be pointed out that the members of a column of P which are related by KVL do not retain that relationship when they are used to form J. For example, we have by KVL that $\frac{P_{1i}}{c_1} - \frac{P_{2i}}{c_2} + \frac{P_{3i}}{c_3} = 0$. When the derivative of this equation is taken with respect to c_1, c_2, or c_3, there is no nice linear relationship between the partial derivatives:

$$\frac{\partial}{\partial c_1}(\frac{P_{1i}}{c_1} - \frac{P_{2i}}{c_2} + \frac{P_{3i}}{c_3}) = (\frac{c_1\frac{\partial P_{1i}}{\partial c_1} - P_{1i}}{c_1^2} - \frac{c_2\frac{\partial P_{2i}}{\partial c_1}}{c_2^2} + \frac{c_3\frac{\partial P_{3i}}{\partial c_1}}{c_3^2}).$$

Cross-Dependencies Suppose no more than two entries are chosen from any column or row of P. It is still possible, though, from *cross-dependencies* for J be singular. Suppose P_{12}, P_{13}, P_{21}, and P_{41} are chosen. Taking the derivative of the equations $P_{11} - P_{12} + P_{13} = 0$ and $P_{11} + P_{21} + P_{41} = 1$ with respect to any c_i gives $\frac{\partial P_{11}}{\partial c_i} - \frac{\partial P_{12}}{\partial c_i} + \frac{\partial P_{13}}{\partial c_i} = 0$ and $\frac{\partial P_{11}}{\partial c_i} - \frac{\partial P_{21}}{\partial c_i} + \frac{\partial P_{41}}{\partial c_i} = 0$. Although P_{11} is not one of the entries selected, the above equations together give $\frac{\partial P_{12}}{\partial c_i} - \frac{\partial P_{13}}{\partial c_i} + \frac{\partial P_{21}}{\partial c_i} + \frac{\partial P_{41}}{\partial c_i} = 0$, a direct dependency between entries in the Jacobian. A Jacobian resulting from a set of entries including $P_{12}, P_{13}, P_{21}, P_{41}$ is singular.

Using Entries From R Suppose entries from R are used in forming the Jacobian. The R matrix is symmetric and has rows and columns related by KVL, as was previously found. The linear relations between rows and between columns are not changed in the Jacobian: $R_{1i} - R_{2i} + R_{3i} = 0$ becomes $\frac{\partial R_{1i}}{\partial c_j} - \frac{\partial R_{2i}}{\partial c_j} + \frac{\partial R_{3i}}{\partial c_j} = 0$, $R_{i1} - R_{i2} + R_{i3} = 0$ becomes $\frac{\partial R_{i1}}{\partial c_j} - \frac{\partial R_{i2}}{\partial c_j} + \frac{\partial R_{i3}}{\partial c_j} = 0$, etc. Therefore, choosing entries which are related by KVL (the corresponding edges form a loop) leads to a singular Jacobian. Also, there are cross-dependencies as there are in P. If we choose $R_{21}, R_{31}, R_{12}, R_{13}$, for example, then

J will be singular because $R_{11} = R_{12} - R_{13} = R_{21} - R_{31}$, and more directly because $R_{21} = R_{12}$ and $R_{13} = R_{31}$. Since R is symmetric, trying to use symmetric entries (R_{ij} and R_{ji}) will always lead to a singular Jacobian in Newton's method.

3.4.1 More About the Jacobian

The idea of the Jacobian being singular because of dependencies within columns of P (or R) leads to some interesting questions:

1. Is there an algorithmic way to determine whether the entries chosen from P or R will lead to a singular Jacobian?

2. Which submatrices of P (or R) contain enough information to solve the inverse problem?

These questions are looked at later in this chapter.

For the 6x6 P matrix of circuit A, no matter which set of six entries are chosen, the resulting Jacobian is singular. That is the reason a constraint must be substituted for one of the equations. Changing all conductances by a multiplicative constant does not change the edge currents, so P remains unchanged. Using Newton's method (or any method), a unique solution to the inverse problem cannot be found from knowledge of P alone. To show that there is always at least one nonzero vector in J's nullspace, a combination of mathematics and physical principles is used. The vector $\vec{c} = (c_1, c_2, c_3, c_4, c_5, c_6)$ is in the nullspace of every Jacobian formed from six entries of P (i.e. $J\vec{c} = 0$). Suppose we have

a Jacobian of the following form:

$$J = \begin{bmatrix} \frac{\partial P_{11}}{\partial c_1} & \frac{\partial P_{11}}{\partial c_2} & \frac{\partial P_{11}}{\partial c_3} & \frac{\partial P_{11}}{\partial c_4} & \frac{\partial P_{11}}{\partial c_5} & \frac{\partial P_{11}}{\partial c_6} \\ \frac{\partial P_{12}}{\partial c_1} & \frac{\partial P_{12}}{\partial c_2} & \frac{\partial P_{12}}{\partial c_3} & \frac{\partial P_{12}}{\partial c_4} & \frac{\partial P_{12}}{\partial c_5} & \frac{\partial P_{12}}{\partial c_6} \\ \frac{\partial P_{13}}{\partial c_1} & \frac{\partial P_{13}}{\partial c_2} & \frac{\partial P_{13}}{\partial c_3} & \frac{\partial P_{13}}{\partial c_4} & \frac{\partial P_{13}}{\partial c_5} & \frac{\partial P_{13}}{\partial c_6} \\ \frac{\partial P_{24}}{\partial c_1} & \frac{\partial P_{24}}{\partial c_2} & \frac{\partial P_{24}}{\partial c_3} & \frac{\partial P_{24}}{\partial c_4} & \frac{\partial P_{24}}{\partial c_5} & \frac{\partial P_{24}}{\partial c_6} \\ \frac{\partial P_{45}}{\partial c_1} & \frac{\partial P_{45}}{\partial c_2} & \frac{\partial P_{45}}{\partial c_3} & \frac{\partial P_{45}}{\partial c_4} & \frac{\partial P_{45}}{\partial c_5} & \frac{\partial P_{45}}{\partial c_6} \\ \frac{\partial P_{66}}{\partial c_1} & \frac{\partial P_{66}}{\partial c_2} & \frac{\partial P_{66}}{\partial c_3} & \frac{\partial P_{66}}{\partial c_4} & \frac{\partial P_{66}}{\partial c_5} & \frac{\partial P_{66}}{\partial c_6} \end{bmatrix}.$$

If $J\vec{c} = 0$ then each row of J must be perpendicular to $\vec{c}$. We prove this for the first and second rows, and it can be proven for the other rows in a similar manner.

To show that the first row of the above J is perpendicular to $\vec{c}$, we need to prove that

$$c_1\frac{\partial P_{11}}{\partial c_1} + c_2\frac{\partial P_{11}}{\partial c_2} + c_3\frac{\partial P_{11}}{\partial c_3} + c_4\frac{\partial P_{11}}{\partial c_4} + c_5\frac{\partial P_{11}}{\partial c_5} + c_6\frac{\partial P_{11}}{\partial c_6} = \vec{c}\cdot\nabla P_{11} = 0. \tag{3.3}$$

Using the formula $P_{11} = c_1\frac{\partial \log\Delta}{\partial c_1}$, we have that

$$c_1\frac{\partial P_{11}}{\partial c_1} = c_1(\frac{\partial \log\Delta}{\partial c_1} + c_1\frac{\partial^2 \log\Delta}{\partial c_1^2}) = P_{11} + c_1^2\frac{\partial^2 \log\Delta}{\partial c_1^2} = P_{11} - P_{11}^2. \tag{3.4}$$

The last step in equation (3.4) follows from the formula $P_{11} = c_1\sqrt{-\frac{\partial^2 \log\Delta}{\partial c_1^2}}$. Again using the fact that $P_{11} = c_1\frac{\partial \log\Delta}{\partial c_1}$, we can rewrite

$$c_2\frac{\partial P_{11}}{\partial c_2} = c_2c_1\frac{\partial^2 \log\Delta}{\partial c_1\partial c_2} = -P_{21}^2\frac{c_1}{c_2}. \tag{3.5}$$

The last step in equation (3.5) comes from the formula $P_{21}^2 = -c_2^2\frac{\partial^2 \log\Delta}{\partial c_1\partial c_2}$. We can rewrite the other terms of equation (3.3) in a similar manner:

$$c_3\frac{\partial P_{11}}{\partial c_3} = -P_{31}^2\frac{c_1}{c_3}, c_4\frac{\partial P_{11}}{\partial c_4} = -P_{41}^2\frac{c_1}{c_4}, c_5\frac{\partial P_{11}}{\partial c_5} = -P_{51}^2\frac{c_1}{c_5}, c_6\frac{\partial P_{11}}{\partial c_6} = -P_{61}^2\frac{c_1}{c_6}.$$

Equation (3.3) can now be written

$$\vec{c}\cdot\nabla P_{11} = P_{11} - P_{11}^2 - P_{21}^2\frac{c_1}{c_2} - P_{31}^2\frac{c_1}{c_3} - P_{41}^2\frac{c_1}{c_4} - P_{51}^2\frac{c_1}{c_5} - P_{61}^2\frac{c_1}{c_6}$$

$$= P_{11} - c_1\left(\frac{P_{11}^2}{c_1} + \frac{P_{21}^2}{c_2} + \frac{P_{31}^2}{c_3} + \frac{P_{41}^2}{c_4} + \frac{P_{51}^2}{c_5} + \frac{P_{61}^2}{c_6}\right).$$

If $c_1 \neq 0$, then the equality to prove becomes

$$\frac{P_{11}}{c_1} - \left(\frac{P_{11}^2}{c_1} + \frac{P_{21}^2}{c_2} + \frac{P_{31}^2}{c_3} + \frac{P_{41}^2}{c_4} + \frac{P_{51}^2}{c_5} + \frac{P_{61}^2}{c_6}\right) = 0. \tag{3.6}$$

The first term $\frac{P_{11}}{c_1}$ is power being put into the circuit by the unit current source. Power is current multiplied by voltage – the current has magnitude one and the voltage across c_1 (where the current is being injected) is $\frac{P_{11}}{c_1}$ by Ohm's Law. The large term in parentheses in equation (3.6) is total power being dissipated in the circuit. Power is current multiplied by voltage and also current squared multiplied by resistance. Thus $\frac{P_{11}^2}{c_1}$ is power being dissipated in c_1, $\frac{P_{21}^2}{c_2}$ is power being dissipated in c_2, $\frac{P_{31}^2}{c_3}$ is power being dissipated in c_3, etc. By **conservation of power**, equation (3.6) must be true: total power in (from source) equals total power out (through resistors). Therefore, $\vec{c}$ is perpendicular to a row of J originating from P_{11}. The above argument can be generalized to show that $\vec{c}$ is perpendicular to any row of J which comes from a *diagonal* element of P.

To show that the second row of the above Jacobian is also perpendicular to c, we must prove that

$$c_1\frac{\partial P_{21}}{\partial c_1} + c_2\frac{\partial P_{21}}{\partial c_2} + c_3\frac{\partial P_{21}}{\partial c_3} + c_4\frac{\partial P_{21}}{\partial c_4} + c_5\frac{\partial P_{21}}{\partial c_5} + c_6\frac{\partial P_{21}}{\partial c_6} = \vec{c}\cdot\nabla P_{21} = 0. \tag{3.7}$$

Using the formula $P_{21}^2 = -c_2{}^2\frac{\partial^2 \log\Delta}{\partial c_2 \partial c_1}$, we find the following:

$$\frac{\partial P_{21}}{\partial c_1} = -\frac{c_2^2}{2P_{21}}\frac{\partial}{\partial c_1}\left(\frac{\partial^2 \log\Delta}{\partial c_2 \partial c_1}\right)$$

$$\frac{\partial P_{21}}{\partial c_2} = -\frac{1}{2P_{21}}(2c_2\frac{\partial^2 \log \Delta}{\partial c_2 \partial c_1} + c_2^2\frac{\partial^3 \log \Delta}{\partial c_2^2 \partial c_1})$$

$$\frac{\partial P_{21}}{\partial c_3} = -\frac{c_2^2}{2P_{21}}\frac{\partial}{\partial c_3}(\frac{\partial^2 \log \Delta}{\partial c_2 \partial c_1})$$

$$\frac{\partial P_{21}}{\partial c_4} = -\frac{c_2^2}{2P_{21}}\frac{\partial}{\partial c_4}(\frac{\partial^2 \log \Delta}{\partial c_2 \partial c_1})$$

$$\frac{\partial P_{21}}{\partial c_5} = -\frac{c_2^2}{2P_{21}}\frac{\partial}{\partial c_5}(\frac{\partial^2 \log \Delta}{\partial c_2 \partial c_1})$$

$$\frac{\partial P_{21}}{\partial c_6} = -\frac{c_2^2}{2P_{21}}\frac{\partial}{\partial c_6}(\frac{\partial^2 \log \Delta}{\partial c_2 \partial c_1}).$$

Substituting these equalities into (3.7) gives

$$\vec{c} \cdot \nabla P_{21} = -\frac{c_2^2}{2P_{21}}(c_1\frac{\partial^3 \log \Delta}{\partial c_1^2 \partial c_2} + [2\frac{\partial^2 \log \Delta}{\partial c_2 \partial c_1} + c_2\frac{\partial^3 \log \Delta}{\partial c_2^2 \partial c_1}] + c_3\frac{\partial^3 \log \Delta}{\partial c_1 \partial c_2 \partial c_3}$$

$$+c_4\frac{\partial^3 \log \Delta}{\partial c_1 \partial c_2 \partial c_4} + c_5\frac{\partial^3 \log \Delta}{\partial c_1 \partial c_2 \partial c_5} + c_6\frac{\partial^3 \log \Delta}{\partial c_1 \partial c_2 \partial c_6}).$$

Since $2\frac{\partial^2 \log \Delta}{\partial c_2 \partial c_1} = -2\frac{P_{21}^2}{c_2^2}$, we have

$$\vec{c} \cdot \nabla P_{21} = P_{21} - \frac{c_2^2}{2P_{21}}(c_1\frac{\partial^3 \log \Delta}{\partial c_1^2 \partial c_2} + c_2\frac{\partial^3 \log \Delta}{\partial c_2^2 \partial c_1} + c_3\frac{\partial^3 \log \Delta}{\partial c_1 \partial c_2 \partial c_3}.$$

$$+c_4\frac{\partial^3 \log \Delta}{\partial c_1 \partial c_2 \partial c_4} + c_5\frac{\partial^3 \log \Delta}{\partial c_1 \partial c_2 \partial c_5} + c_6\frac{\partial^3 \log \Delta}{\partial c_1 \partial c_2 \partial c_6})$$

Again rewriting gives

$$\vec{c} \cdot \nabla P_{21} = P_{21} - \frac{c_2^2}{2P_{21}}\{\frac{\partial}{\partial c_2}(-\frac{P_{11}^2}{c_1}) + c_2\frac{\partial}{\partial c_2}(-\frac{P_{21}^2}{c_2^2}) + \frac{\partial}{\partial c_2}(-\frac{P_{31}^2}{c_3}) + \frac{\partial}{\partial c_2}(-\frac{P_{41}^2}{c_4})$$

$$+\frac{\partial}{\partial c_2}(-\frac{P_{51}^2}{c_5}) + \frac{\partial}{\partial c_2}(-\frac{P_{61}^2}{c_6})\}.$$

Using the fact that

$$c_2\frac{\partial}{\partial c_2}(-\frac{P_{21}^2}{c_2^2}) = \frac{\partial}{\partial c_2}(-\frac{P_{21}^2}{c_2}) + \frac{P_{21}^2}{c_2^2},$$

we arrive at the equation

$$\vec{c}\cdot\nabla P_{21} = \frac{P_{21}}{2} + \frac{c_2^2}{2P_{21}}\frac{\partial}{\partial c_2}(\frac{P_{11}^2}{c_1} + \frac{P_{21}^2}{c_2} + \frac{P_{31}^2}{c_3} + \frac{P_{41}^2}{c_4} + \frac{P_{51}^2}{c_5} + \frac{P_{61}^2}{c_6})$$

$$= \frac{c_2^2}{2P_{21}}\{\frac{P_{21}^2}{c_2^2} + \frac{\partial}{\partial c_2}(\frac{P_{11}^2}{c_1} + \frac{P_{21}^2}{c_2} + \frac{P_{31}^2}{c_3} + \frac{P_{41}^2}{c_4} + \frac{P_{51}^2}{c_5} + \frac{P_{61}^2}{c_6})\}. \tag{3.8}$$

We recognize the terms inside the derivative as total power dissipated in the circuit, which is equal to total power injected into the circuit $\frac{P_{11}}{c_1}$. Then

$$\frac{\partial}{\partial c_2}(\frac{P_{11}}{c_1}) = \frac{\partial}{\partial c_2}(\frac{\partial \log \Delta}{\partial c_1}) = -\frac{P_{21}^2}{c_2^2}$$

shows that (3.8) is equal to zero, i.e.

$$\vec{c}\cdot\nabla P_{21} = \frac{c_2^2}{2P_{21}}(\frac{P_{21}^2}{c_2^2} - \frac{P_{21}^2}{c_2^2}) = 0.$$

Therefore $\vec{c}$ is perpendicular to the second row of J.

The remaining rows of J can be shown to be perpendicular to $\vec{c}$ by following the same argument as for row 2. In fact, the argument can be used to show that $\vec{c}\cdot\nabla P_{ij} = 0$ for any i,j. Thus, any Jacobian matrix formed exclusively from entries of P (no constraint equation) will contain $\vec{c}$ in its nullspace and therefore be singular. To obtain a unique answer using Newton's method, a constraint equation is used in place of one of the equations from P. Five independent entries of P and a constraint equation (like $\prod c_i = 1$) produce a nonsingular Jacobian for the inverse problem of circuit A.

3.4.2 The Algorithm: Is J Singular?

Up to this point, we have looked only at the specific case of circuit A. We now step back from that specific case to draw some general conclusions. When solving the inverse problem via Newton's method for any network, it is helpful to have an algorithm to determine whether the measurements taken (entries of P or R observed) lead to a singular Jacobian.

We have seen that the key in the specific case is KVL and KCL, and this should remain true for other circuits in general. For an m-edge circuit, any m current measurements (entries of P) lead to a singular Jacobian: $\vec{c} = (c_1, c_2, ..., c_m)$ is perpendicular to the vector $(\frac{\partial P_{ij}}{\partial c_1}, \frac{\partial P_{ij}}{\partial c_2}, ..., \frac{\partial P_{ij}}{\partial c_m})$ for any i,j. This can be shown using power formulas as in the above derivation for circuit A. One would like to know which sets of $m-1$ entries from P (along with the constraint $\prod c_i = 1$) lead to a nonsingular Jacobian. For R, one would like to know which sets of m measurements lead to a nonsingular Jacobian.

The following algorithm is proposed to determine if the $m-1$ chosen entries from P (or m from R) lead to a singular Jacobian. After the algorithm is explained, an example of its use is presented. The goal is to provide a systematic way of determining if there are any subsets of the chosen entries which would cause the resulting Jacobian to be singular. The algorithm can be divided into several steps:

1. Circle the entries of P or R corresponding to measurements taken.

2. Determine if entries falling in a single row or column of P or R lead to dependencies in J. For the P matrix, dependencies among members of a column which lead to a singular J are found by KCL, and such dependencies among members of a row are found by KVL. For R, the dependencies which lead to a singular J are determined by KVL. NOTE: Elements in each column of P are also related by KVL, but those dependencies do not survive in the Jacobian and so do not enter into this algorithm (see p.32).

3. If no dependencies are found in step 2, then identify all "virtual" measurements and draw boxes around them. "Virtual" measurements are here defined to be entries whose values can be determined (via the KCL or KVL relations of step 2) from the circled measurements in a single row or column of the matrix. If a virtual entry is created twice (once by entries of row and once by entries of a column), then put two boxes around it. If no virtual entries are created by the original (circled) entries, then the original entries can be used to form a nonsingular J.

4. If a virtual entry is boxed twice, then the original entries of P or R which created that entry (in the previous step) will lead to a singular J. That is because the double box indicates cross-dependencies among those original entries (see p.32). If no virtual entries are boxed at least twice, then proceed to the next step.

5. Find all "second-level" virtual entries – entries whose values can be found from marked entries (using the same KVL and KCL relations as in step 2) — and use a third shape, say diamonds, to identify them. If there are no such "second-level" entries, then the algorithm is done, and the original entries lead to a nonsingular Jacobian. If any second-level virtual entries are created, go on to the next step.

6. If a second-level entry is marked twice, trace back and find all original (circled) entries which led to the twice-marked entry. If all of those original entries form a single submatrix of P or R, then remove one of the marks from the second-level virtual entry (this will be made clear in the example of Figure 3-1a). If the original entries do not constitute a submatrix of P or R, then they lead to a singular Jacobian, and the algorithm is done. If no second-level virtual entries are marked twice, then go on to the next step.

7. Keep repeating the last two steps for higher and higher levels using a different way to identify entries at each level. At some stage either a virtual entry will be marked twice (indicating its parent entries lead to a singular Jacobian) or there are no new virtual entries (indicating that the original entries do not lead to a singular Jacobian).

This algorithm has not been proven or extensively tested, and is therefore suspect. The main pitfall seems to be in step 6, where one has to check if certain original entries form a submatrix of P or R. The idea is that the entries in a submatrix, unless found dependent in step 2, should not have any further dependencies. In Figure 3-1a, if columns 1,2,3 and rows 1,2,3 are related by KVL or KCL, then the original entries lead to a twice marked entry in the upper left, as shown. The original entries form a submatrix, though, and

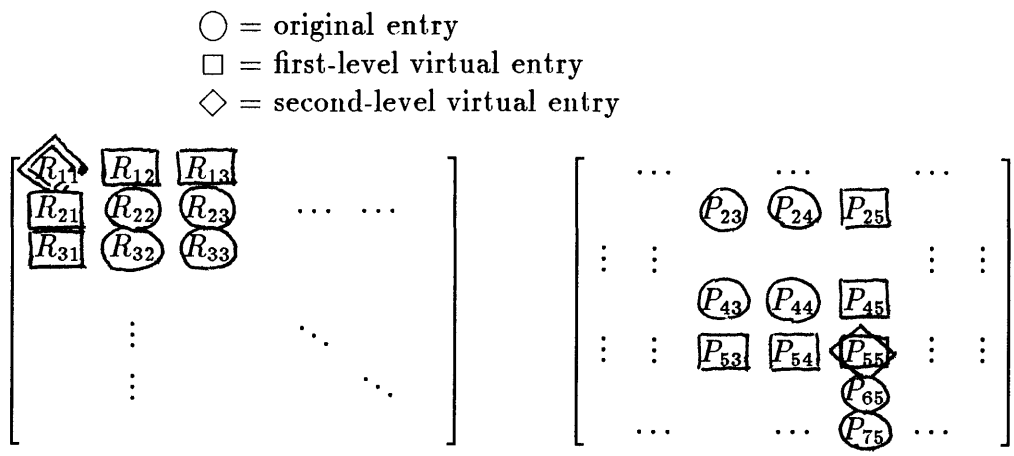


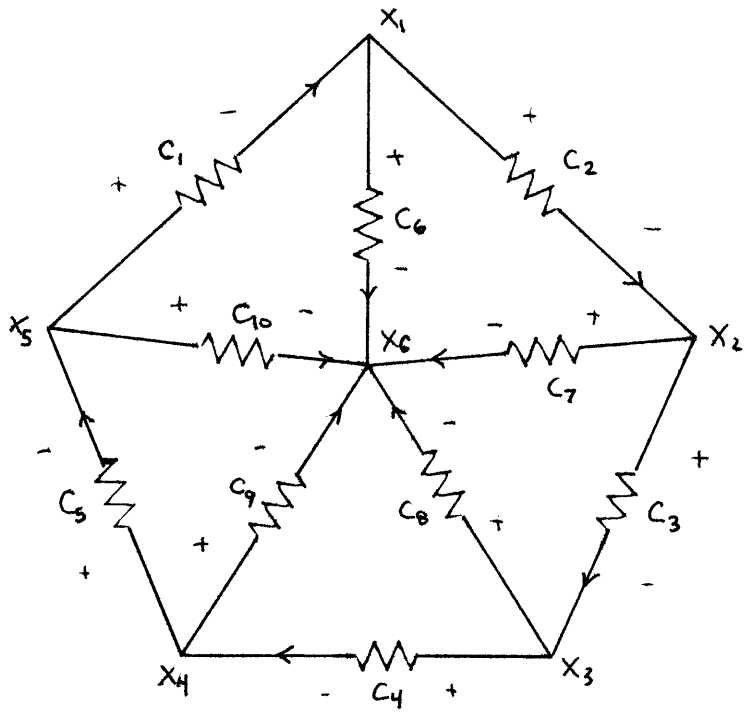
Figure 3-1: (a) Original entries form submatrix. (b) Marked at two levels

so do not lead to a singular Jacobian. The underlying idea is that each of the original entries was used twice (once row-wise and once column-wise) to form the second-level virtual entry, so that the second-level entry should really only be marked once. A virtual entry which is marked twice, even at two different levels as in Figure 3-1b, indicates a singular Jacobian (unless the parent entries form a submatrix, as noted above). One can identify all the submatrices formed by original entries. After step 2, it is only relations *between* submatrices (not those caused by a single submatrix) which can cause the the Jacobian to be singular.

A complete example is shown in Figure 3-2. The nine current measurements

$$P_{11}, P_{51}, P_{16}, P_{56}, P_{73}, P_{75}, P_{10,5}, P_{10,9}, P_{35}$$

are taken and the entries corresponding to those measurements are circled in the 10×10 P matrix. One can write down KCL and KVL dependencies from the nodes and loops of the circuit respectively: from node 1 we have that column entries 1,2,6 are dependent, from node 2 we have that column entries 2,3,7 are dependent, etc. The loops of the circuit indicate which row entries are dependent. For example, the loop formed by edges 1,6,10 indicate that row entries 1,6,10 are dependent. In stage 2, we check the circled entries and find that none of the above relations indicate a singular J matrix. In stage 3,



○ = original entry
□ = first-level virtual entry
◇ = second-level virtual entry

$$\begin{bmatrix} P_{11} & \cdots & & P_{16} & & P_{1,10} \\ & \ddots & P_{25} & & & \\ \vdots & \ddots & P_{35} & & \vdots & \vdots \\ P_{51} & & & P_{56} & & P_{5,10} \\ \vdots & P_{73} & P_{75} & & \vdots & \vdots \\ P_{10,1} & & P_{10,5} & P_{10,6} & P_{10,9} & P_{10,10} \end{bmatrix}$$

Figure 3-2: Circuit and P matrix

we put square boxes around the "virtual" entries. Entries $P_{10,5}$ and $P_{10,9}$ can be used to determine $P_{10,10}$ by KVL, so that entry gets a square box. Entries P_{11} and P_{51} determine the value of $P_{10,1}$ by KCL, so that entry also gets a square box. The rest of the virtual entries are also found by the KVL and KCL relations. In stage 4, we check and find that no virtual entries are marked twice. In stage 5, we create second-level virtual entries from first-level virtual entries. This is done exactly the same way we created virtual entries in stage 3. The virtual entries $P_{10,1}$ and $P_{10,6}$ lead to the second-level entry $P_{10,10}$. The virtual entries $P_{1,10}$ and $P_{5,10}$ also lead to $P_{10,10}$, so we put another box around it. At this point, $P_{10,10}$ has been marked three times – twice at the second level and once at the first level. The parent entries $P_{11}, P_{16}, P_{51}, P_{56}$ form a rectangle, however, so one of the second level marks (diamond boxes) is removed. The 10,10 entry is still marked twice, though, and so the algorithm has found that the parent entries $P_{11}, P_{16}, P_{51}, P_{56}, P_{10,5}, P_{10,9}$ lead to a singular Jacobian. By KCL, we have that $1 + P_{51} - P_{11} = P_{10,1}$ and $P_{56} - P_{16} = P_{10,6}$. By KVL, we have that $P_{10,1} + P_{10,6} = P_{10,10}$ and $P_{10,9} - P_{10,5} = P_{10,10}$. (The signs result from the edge directions chosen in Figure 3-2.) Therefore, the algebraic dependency between the parent entries is

$$1 + P_{51} - P_{11} + P_{56} - P_{16} - P_{10,9} + P_{10,5} = 0.$$

3.4.3 Solving From a Submatrix of P or R

In this section, we deal with a general m-edge, n-node circuit and ask what size submatrix of the $m \times m$ P or R matrices contains enough information to solve the inverse problem. To solve for the m edge conductances, we need $m - 1$ entries of P and one constraint equation, or m entries from R.

The independent loop currents in the circuit determine the left nullspaces of P and R ($P^T\vec{y} = 0, R^T\vec{y} = 0$). There are $m - n + 1$ independent loops in the circuit ([6], p.119) and hence $m - n + 1$ vectors form a basis for the left nullspaces of P and R. A set of independent loop currents is formed from the "mesh loops" of a planar graph. Figure

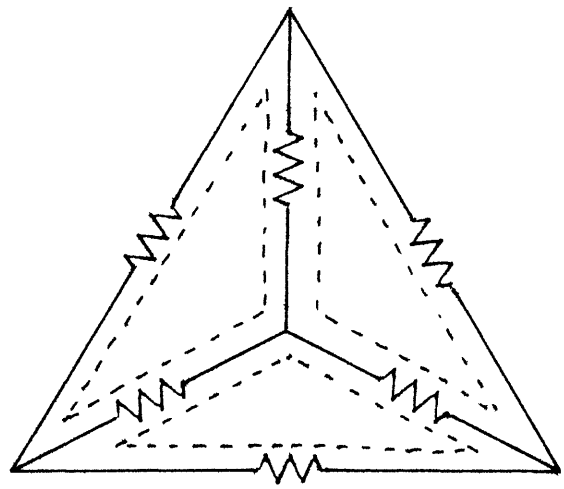


Figure 3-3: Mesh loops of circuit A

3-3 shows the mesh loops of circuit A. The sum of the dimensions of the column space and the left nullspace of P equals m. The same is true for R. Therefore the rank of both P and R is $m - (m - n + 1) = n - 1$. There are $n - 1$ linearly independent rows and independent columns.

From each row of P or R, we can find $n - 1$ algebraically independent entries and the same is true for each column. Considering R, if we take $n - 1$ independent entries from certain rows, we need $m/(n - 1)$ (rounded up to an integer) such rows to have enough information to solve the inverse problem. For P, if we take $n - 1$ independent entries from certain rows, we need $(m - 1)/(n - 1)$ (rounded up to an integer) such rows to have enough information to solve the inverse problem. The dimensions of these submatrices ($(n - 1) \times (\frac{m}{n-1})$ for R, and $(n - 1) \times (\frac{m-1}{n-1})$ for P) can vary as long as neither dimension exceeds $n - 1$ and the entries chosen are not dependent.

The rows and columns of a submatrix correspond to edges in the circuit. For example, in the submatrix of R shown in the upper left of Figure 3-4, the rows correspond to edges 1 and 2 of the circuit, and the columns correspond to edges 1,2,4. For a submatrix of R, if the set of edges corresponding to the rows or the set corresponding to the columns forms

a loop, then those rows or columns corresponding to the edges of the loop are dependent by KVL (some linear combination of those rows or columns gives the zero vector). The submatrix then does not contain enough information to solve the inverse problem.

For a submatrix of P, if the set of edges corresponding to columns form a loop, then those columns are linearly dependent by KVL. If the set of edges corresponding to the rows include all the edges stemming from a single node of the circuit, then entries within each column of the submatrix are algebraically dependent by KCL. (The rows of the submatrix are not necessarily linearly dependent, though, since the KCL equations may be slightly different for each column.) If one of the above dependencies is found in the submatrix of P, then the entries do not contain enough information to solve the inverse problem within a scale factor.

Once again using circuit A as an example, the submatrices of R and P shown on the left in Figure 3-4 each contain enough information to solve the inverse problem, whereas the submatrices of R and P shown on the right do not. The submatrix in the upper right has columns corresponding to edges 1,2,3 of circuit A. Those edges form a loop, so the columns of the submatrix are linearly dependent by KVL. The submatrix on the middle right-hand side of Figure 3-4 has rows corresponding to edges 1,4,5 of the circuit. Since those edges form a loop, the rows of the submatrix are dependent by KVL. In the case shown on the lower right, the rows of the submatrix correspond to edges 4,5,6. Those edges include all edges coming from node 4 of circuit A, so the entries in each column of the submatrix are algebraically dependent by KCL.

Certain submatrices (sets of measurements) may be appropriate in certain practical cases. For example, using a submatrix with as few columns as possible might be advantageous if sources are costly or if access to only certain edges is possible (remember that each column of P and R corresponds to measurements which can be taken using a source across one edge). In other cases, using many sources might present no problems, and a submatrix with more columns can be used.

Enough information	Not enough information
$\begin{bmatrix} R_{11} & R_{12} & & R_{14} \\ R_{21} & R_{22} & & R_{24} \\ & & & \\ & & & \\ & & & \end{bmatrix}$	$\begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ & & \\ & & \\ & & \end{bmatrix}$
$\begin{bmatrix} & R_{12} & R_{13} \\ & R_{32} & R_{33} \\ & R_{52} & R_{53} \end{bmatrix}$	$\begin{bmatrix} R_{11} & R_{12} \\ & \\ R_{41} & R_{42} \\ R_{51} & R_{52} \\ & \end{bmatrix}$
$\begin{bmatrix} P_{11} & P_{13} & P_{15} \\ P_{31} & P_{33} & P_{35} \\ & & \\ & & \end{bmatrix}$	$\begin{bmatrix} & \\ & \\ P_{41} & P_{43} \\ P_{51} & P_{53} \\ P_{61} & P_{63} \end{bmatrix}$

Figure 3-4: Various submatrices of R and P

Conclusion

In this thesis, we examined the inverse problem for electrical networks. We have presented an algorithm to determine whether certain source/measurement pairs contain enough information to solve the inverse problem. This algorithm is simplified for certain sets of data, corresponding to submatrices of the P and R matrices. One area for future work is the implementation of this algorithm on a computer. A useful program would take as input the connectivity matrix of a network, and use that information to determine which measurements are sufficient to solve the inverse problem.

We have presented computer programs which solve the inverse problem for specific cases. A set of nonlinear equations for the conductances is solved by Newton's method and separately by an improved algorithm which applies Newton's method at each iteration.

We have also shown that the fundamental requirements for existence of solutions to the inverse problem can be found from physical laws, as well as from graph theory. The physical laws which we have used in this thesis are Kirchhoff's Current Law and Kirchoff's Voltage Law. It is a reasonable conjecture that in other kinds of networks (mechanical, for example), requirements for existence of solutions to the inverse problem can also be determined from physical laws.

Appendix A

Sensitivity to Changes in Conductances

For circuits with nearly equal conductances, there is a simple way to find the effect of small changes in the conductances on the measurements in P or R. The surfaces described by the equations for the entries of P or R can be locally approximated by tangent planes. By using these tangent plane approximations, the effects on P and R of small changes in the conductances can be calculated.

We illustrate the method for the P matrix of circuit A (Figure 1-1). If the conductances are approximately equal, then we can write $c_1 = c_0 + \Delta c_1, c_2 = c_0 + \Delta c_2, ..., c_6 = c_0 + \Delta c_6$ where c_0 is some value close to all the c's (perhaps an average value). Because scaling all the conductances by the same multiplicative constant does not change P, we can divide all conductances by c_0 to simplify the problem: $c_1 = 1 + \delta c_1, c_2 = 1 + \delta c_2, ..., c_6 = 1 + \delta c_6$ where $\delta c_i = \frac{\Delta c_i}{c_0}$. Substituting these expressions into the equation for P_{11} gives:

$$P_{11} = \frac{1}{2} + \delta d_{11} = \frac{(1+\delta c_1)(1+\delta c_2)(1+\delta c_4) + \ldots 7 \textit{ other products}}{(1+\delta c_1)(1+\delta c_2)(1+\delta c_4) + \ldots 15 \textit{ other products}}$$

where δd_{11} is the change in P_{11} from its value at $(c_1, c_2, c_3, c_4, c_5, c_6) = (1, 1, 1, 1, 1, 1)$. By throwing out higher order terms (products of the δc's) and using standard first order

approximations, we arrive at equation (A.1).

$$\delta d_{11} \approx \frac{1}{4}\delta c_1 - \frac{1}{16}(\delta c_2 + \delta c_3 + \delta c_4 + \delta c_5). \tag{A.1}$$

This equation allows one to directly find the change in P_{11} resulting from small changes in the conductances. Similar equations can be found for the other entries of P.

For any six (or five) entries of P chosen to use in Newton's method, the vector equations like (A.1) can be assembled into a matrix equation. For example, suppose we choose the diagonal terms $P_{11}, P_{22}, ..., P_{66}$. Then the matrix equation is:

$$\begin{bmatrix} \delta d_{11} \\ \delta d_{22} \\ \delta d_{33} \\ \delta d_{44} \\ \delta d_{55} \\ \delta d_{66} \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & -\frac{1}{16} & -\frac{1}{16} & -\frac{1}{16} & -\frac{1}{16} & 0 \\ -\frac{1}{16} & \frac{1}{4} & -\frac{1}{16} & -\frac{1}{16} & 0 & -\frac{1}{16} \\ -\frac{1}{16} & -\frac{1}{16} & \frac{1}{4} & 0 & -\frac{1}{16} & -\frac{1}{16} \\ -\frac{1}{16} & -\frac{1}{16} & 0 & \frac{1}{4} & -\frac{1}{16} & -\frac{1}{16} \\ -\frac{1}{16} & 0 & -\frac{1}{16} & -\frac{1}{16} & \frac{1}{4} & -\frac{1}{16} \\ 0 & -\frac{1}{16} & -\frac{1}{16} & -\frac{1}{16} & -\frac{1}{16} & \frac{1}{4} \end{bmatrix} \begin{bmatrix} \delta c_1 \\ \delta c_2 \\ \delta c_3 \\ \delta c_4 \\ \delta c_5 \\ \delta c_6 \end{bmatrix}. \tag{A.2}$$

This matrix equation allows one to find the variations in the current probe data given the variations in the conductances. To solve the inverse problem, i.e. find the δc's from the δd's, one must invert the matrix in (A.2). This matrix, let us call it B, is singular, however, because the columns add to give the zero vector. There is an arbitrary additive constant in the δc's. Adding a multiple of (1,1,1,1,1,1) to a solution leaves the δd's unchanged, since the vector (1,1,1,1,1,1) is in the nullspace of B. We can replace one of the P_{jj}, say P_{66}, with the constraint $\prod c_i = 1$. Using the same substitution ($c_i = 1 + \delta c_i$) as above, and throwing out higher order terms, we arrive at

$$\delta c_1 + \delta c_2 + \delta c_3 + \delta c_4 + \delta c_5 + \delta c_6 \approx 0.$$

Using this approximation, we can rewrite the matrix equation (A.2) as the following:

$$\begin{bmatrix} \delta d_{11} \\ \delta d_{22} \\ \delta d_{33} \\ \delta d_{44} \\ \delta d_{55} \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & -\frac{1}{16} & -\frac{1}{16} & -\frac{1}{16} & -\frac{1}{16} & 0 \\ -\frac{1}{16} & \frac{1}{4} & -\frac{1}{16} & -\frac{1}{16} & 0 & -\frac{1}{16} \\ -\frac{1}{16} & -\frac{1}{16} & \frac{1}{4} & 0 & -\frac{1}{16} & -\frac{1}{16} \\ -\frac{1}{16} & -\frac{1}{16} & 0 & \frac{1}{4} & -\frac{1}{16} & -\frac{1}{16} \\ -\frac{1}{16} & 0 & -\frac{1}{16} & -\frac{1}{16} & \frac{1}{4} & -\frac{1}{16} \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \delta c_1 \\ \delta c_2 \\ \delta c_3 \\ \delta c_4 \\ \delta c_5 \\ \delta c_6 \end{bmatrix}.$$

The B matrix previously had rank 5 and the modified B, call it B_0, has full rank and is invertible. The δc's can be found from the δd's. The approximation to the constraint forces the δc's to add to zero.

Notice that the symmetries of the circuit manifest themselves in the B matrix. The unmodified B matrix is symmetric. The element $P_{11} = \frac{1}{2} + \delta d_{11}$ is most sensitive to changes in c_1 and least sensitive to changes in c_6. The sensitivities of the other diagonal elements may also be found by looking at the B matrix: each element is most sensitive to changes in the conductance which has the current source, and least sensitive to the conductance which is symmetrically opposite the edge with the current source across it. These sensitivities change as we move away from $(c_1, c_2, c_3, c_4, c_5, c_6) = (1, 1, 1, 1, 1, 1)$ and $(P_{11}, P_{22}, \ldots, P_{66}) = (\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2})$.

It is important to point out that B is equivalent to the Jacobian matrix (formed from the diagonal elements of P for the above case) evaluated at $(c_1, c_2, c_3, c_4, c_5, c_6) = (1, 1, 1, 1, 1, 1)$. This is logical, since by calculus we can write approximations like the following for the changes in the diagonal elements of P:

$$\delta d_{11} \approx \frac{\partial P_{11}}{\partial c_1}\delta c_1 + \frac{\partial P_{11}}{\partial c_2}\delta c_2 + \ldots + \frac{\partial P_{11}}{\partial c_6}\delta c_6$$

(where δd_{11} is the change in P_{11}). The substitutions and approximations used at the beginning of Appendix A allow one to find the above partial derivatives (locally) without

actually taking derivatives. This approach may be useful in certain cases.

Appendix B

Computer Programs

B.1 NEWTON3

```
C                                NEWTON3
C
C       This program uses Newton's method to solve the inverse resistance
C       problem for a ring of three resistors.
C
C234567-----------------------------------------------------------------|------
C
C       inputs will be d(1), d(2), d(3) which are the fractions of
C         current flowing across each edge due to a unit current probe
C         across that edge.
C       inputs will also include c(1), c(2), c(3) which are the first
C         guesses for the conductance on each edge (c(1)c(2)c(3))=1 is
C         one of the constraints in order to give a unique solution.
```

```
C       program stops when it gets within a certain tolerance of 0 using
C         Newton's method.
C
      integer ia, n, iaa, ifail,j,k
      real*8 a(10,10), b(10), x(10), aa(10,10), wks1(10), wks2(10)
      real*8 d(3), c(3), delta, delsquare
      external F04ATF
C      Get data from current probes and initial guess
      print *, 'Please input the data from the edges'
      print *, 'in order, one at a time'
      read *, d(1)
      read *, d(2)
      read *, d(3)
      print *, 'Now input your initial guesses for the conductances'
      print *, 'in order, one at a time'
      read *, c(1)
      read *, c(2)
      read *, c(3)
C      Now run the Newton's method loop.
C      Loop will run until either right-hand side becomes < (.001,.001,
C      .001) (absolute value) or until 100 iterations have been run.
C      j will be the counter for the number of loops
      j=0
C      Set up parameters for Ax=b subroutine
      ia = 10
      iaa = 10
      n = 3
      ifail = 0
```

```
C       Check if right-hand side is within required closeness to 0
 10    delta = c(1)*c(2) + c(2)*c(3) + c(3)*c(1)
       delsquare = delta**2
       b(1) = (delta-c(2)*c(3))/delta - d(1)
       b(2) = (delta-c(1)*c(3))/delta - d(2)
       b(3) = c(1)*c(2)*c(3) - 1
       if (abs(b(1)).LE.(.001).and.abs(b(2)).LE.(.001).and.
      + abs(b(3)).LE.(.001)) goto 50
C       otherwise set up matrix equation and perform Newton's method
       do 15 k = 1,3
          b(k) = -1*delta*b(k)
 15       continue
       a(1,1) = (c(3)*c(2)**2+c(2)*c(3)**2)/delta
       a(1,2) = -1*c(1)*c(3)**2/delta
       a(1,3) = -1*c(1)*c(2)**2/delta
       a(2,1) = -1*c(2)*c(3)**2/delta
       a(2,2) = (c(3)*c(1)**2+c(1)*c(3)**2)/delta
       a(2,3) = -1*c(2)*c(1)**2/delta
       a(3,1) = c(2)*c(3)*delta
       a(3,2) = c(1)*c(3)*delta
       a(3,3) = c(1)*c(2)*delta
       call F04ATF(a, ia, b, n, x, aa, iaa, wks1, wks2, ifail)
C       ifail will cause a break here if A matrix was close to singular
C       otherwise update values of c and perform another iteration
C         (only if j<100)
       j = j+1
       if (j.EQ.100) goto 50
       do 20 k = 1,3
```

```
        c(k) = x(k) + c(k)
 20     continue
      goto 10
C      This is the output portion of the program
C      We can only get to this point if 100 loops have been performed
C       or if one of the intermediate solutions is good enough
 50    if (j.EQ.100) then
         print *, 'The result after 100 iterations was...'
       else
         print *, 'The result after ',j,' iterations was...'
       endif
       print *, 'conductance #1 equals  ',c(1)
       print *, 'conductance #2 equals  ',c(2)
       print *, 'conductance #3 equals  ',c(3)
       end
```

B.2 NEWTON6A

```
C                                NEWTON6A
C
C       This program uses Newton's method to solve the inverse resistance
C       problem for a particular network of six resistors and four nodes.
C       First we try an initial guess, and if that doesn't work,
C       then we creep up on the actual current fractions by starting with
C       a set of fractions for which we know the answer, and slowly
C       varying this set toward the desired set of current fractions.
C       Outputs are iteratively fed back into the algorithm.  The best
C       scheme for making the steps toward the desired current fractions
C       is not yet determined.
C
C234567-----------------------------------------------------------------|------
C
C       inputs will be d(1),d(2),...,d(6) which are the fractions of
C         current flowing across each edge due to a unit current probe
C         across that edge.
C       c(1),c(2),...,c(6) will end up as the
C        values of the conductances on each edge.  c(1)c(2)...c(6)=1 is
C         one of the constraints in order to give a unique solution.
C       e(1),e(2),...,e(6) hold the intermediate current fractions en
C         route to the answer (if the initial guess doesn't work).
C       program stops when it gets within a certain tolerance of 0 using
C         Newton's method.
```

```
C
      integer ia, n, iaa, ifail,j,k,try,flag,done
      real*8 a(10,10), b(10), x(10), aa(10,10), wks1(10), wks2(10)
      real*8 d(6), c(6), delta, delsquare, e(6), step,l
      real*8 sum, diff, esave(6), csave(6)
      external F04ATF
      try=1
C      Get data from current probes and set all conductances to 1
C      These measurements correspond to the diagonal elements of P
      print *, 'Please input the data from the edges'
      print *, 'in order, one at a time'
 1    do 2 k=1,6
        read *, d(k)
 2      continue
C      make sure current fractions add up to 3
C      this is a requirement on the diagonal elements of P
      if ((d(1)+d(2)+d(3)+d(4)+d(5)+d(6)).GE.(3.01).or.
     + (d(1)+d(2)+d(3)+d(4)+d(5)+d(6)).LE.(2.99)) then
         print *, 'Fractions add to ',d(1)+d(2)+d(3)+d(4)+d(5)+d(6)
         print *, 'Please reenter the data'
         goto 1
      endif
C
C      First we try the particular initial guess equal to the d's
C      If it doesn't work, then we use the alternate method.
C
C      for the initial guess, we set the conductances equal to the
C      current fractions.
```

```
      do 200 k=1,6
         c(k)=d(k)
 200  continue
C      next set up to perform Newton's method
      j=0
      ia=10
      iaa=10
      n=6
      ifail=0
C     compute delta and delta squared and check if too big
 210  delta = c(1)*c(2)*c(4) + c(1)*c(2)*c(5) + c(1)*c(2)*c(6)+
     + c(1)*c(3)*c(4) + c(1)*c(3)*c(5) + c(1)*c(3)*c(6) + c(1)*c(5)*c(6)
     + + c(1)*c(4)*c(6) + c(2)*c(3)*c(5) + c(2)*c(3)*c(6)+c(2)*c(3)*c(4)
     + + c(2)*c(5)*c(6)+c(2)*c(5)*c(4)+c(3)*c(4)*c(5)+c(3)*c(4)*c(6)
     + + c(4)*c(5)*c(6)
      delsquare = delta**2
C      print *, delsquare
C      if delta squared is too big, Newton's method is probably
C      diverging, so try alternate method.
      if (delsquare.ge.(1d+20)) then
         print *, 'Initial guess failed, using alternate method'
         print *, 'loop #',j
         goto 3
      endif
      done=0
C      the setup subroutine calculates the values for the Jacobian
C      and checks to see if a solution is found.  If a solution is found
C      the subroutine sets done=1
```

```
      call setup(a,b,c,d,6,10,delta,delsquare,done)
      if (done.EQ.1) goto 250
      call F04ATF(a, ia, b, n, x, aa, iaa, wks1, wks2, ifail)
      j=j+1
      if (j.EQ.100) then
         print *, 'Initial guess failed...'
         print *, 'loop #',j
         goto 3
      endif
C      if no solution is found, update the guess and try again
      do 220 k=1,6
         c(k)=x(k)+c(k)
 220     continue
      goto 210
C     Output portion -- only get here if initial guess worked
 250  print *, 'The initial guess worked.'
      print *, 'The answer is...'
      do 255 k=1,6
         print *, 'conductance #',k,' equals ',c(k)
 255     continue
      goto 110
C
C    We get to this portion of the program if the initial guess failed
C     This is the alternate method.
C
 3    do 4 k=1,6
        c(k)=1
        e(k)=.5
```

```
 4      continue
C       At this point we set up the intermediate problem (with the e's)
C       and solve that.
C       We step logarithmically toward the desired fractions.
       l=-8
 5     step=2**l
       flag=0
C       save the intermediate fractions and c's before we update them.
C        then if it blows up, we can go back and use smaller steps from
C        the last solution that worked (we were too greedy).
       do 6 k=1,6
          esave(k)=e(k)
          csave(k)=c(k)
 6        continue
 39    do 7 k=1,6
          e(k)=e(k)+step*(d(k)-e(k))
 7        continue
C       now adjust the e's so that they add to 3
C       (if they don't add to 3, funny things may happen)
       diff = e(1)+e(2)+e(3)+e(4)+e(5)+e(6)-3
       sum = abs(e(1)-d(1))+abs(e(2)-d(2))+abs(e(3)-d(3))+
      + abs(e(4)-d(4))+abs(e(5)-d(5))+abs(e(6)-d(6))
       if (abs(diff).gt.01) then
          do 8 j=1,6
             e(j)=e(j)- abs(e(j)-d(j))*diff/sum
 8           continue
       endif
C       print out intermediate guess  (for error checking)
```

```
C      print *,'Intermediate fractions ',e(1),e(2),e(3),e(4),e(5),e(6)
C      Now run the Newton's method loop.
C      Loop will run until either right-hand side becomes < (.001,.001,
C      ...,.001) (absolute value) or until 100 iterations have been run.
C      j will be the counter for the number of loops
      j=0
C      Set up parameters for Ax=b subroutine
      ia = 10
      iaa = 10
      n = 6
      ifail = 0
C     compute delta and delta squared and check if too big
 10   delta = c(1)*c(2)*c(4) + c(1)*c(2)*c(5) + c(1)*c(2)*c(6)+
     + c(1)*c(3)*c(4) + c(1)*c(3)*c(5) + c(1)*c(3)*c(6) + c(1)*c(5)*c(6)
     + + c(1)*c(4)*c(6) + c(2)*c(3)*c(5) + c(2)*c(3)*c(6)+c(2)*c(3)*c(4)
     + + c(2)*c(5)*c(6)+c(2)*c(5)*c(4)+c(3)*c(4)*c(5)+c(3)*c(4)*c(6)
     + + c(4)*c(5)*c(6)
      delsquare = delta**2
C      print *, 'delsquare is ',delsquare
C      here we place some sort of check to see if the initial guess is
C        diverging rapidly.  If delsquare
C        is bigger than 10 to the 20th, something is probably wrong.
      if (delsquare.gt.(1d+20)) then
C         use smaller steps from the last set of fractions that worked
         do 11 k=1,6
            e(k)=esave(k)
 11         c(k)=csave(k)
         try=try+1
```

```
         if (try.eq.10) goto 100
         l=-8
         step=2**l
         if (flag.eq.0) then
             flag=1
             goto 39
         else
             try=10
             goto 100
         endif
      endif
C     call the subroutine which sets up the matrices for Newton
      done=0
      call setup(a,b,c,e,6,10,delta,delsquare,done)
C     if done=1 Newton's method has found an intermediate solution
      if (done.eq.1) goto 50
      call F04ATF(a, ia, b, n, x, aa, iaa, wks1, wks2, ifail)
C      ifail will cause a break here if A matrix was close to singular
C      otherwise update values of c and perform another iteration
C       (only if j<100)
      j = j+1
      if (j.EQ.100) goto 50
      do 20 k = 1,6
         c(k) = x(k) + c(k)
 20      continue
      goto 10
C      This is the output portion of the program
C      We can only get to this point if 100 loops have been performed
```

```
C       or if one of the intermediate solutions is good enough
 50   if (abs(l).LE.(.001).or.(abs(e(1)-d(1)).le.(.0001).and.
     +  abs(e(2)-d(2)).le.(.0001).and.abs(e(3)-d(3)).le.(.0001).and.
     +  abs(e(4)-d(4)).le.(.0001).and.abs(e(5)-d(5)).le.(.0001).and.
     +  abs(e(6)-d(6)).le.(.0001))) then
         print *, 'the solution to within a scale factor is...'
         do 55 k=1,6
            print *, 'Conductance #',k,' is ',c(k)
 55         continue
      else
         l=l+1
         goto 5
      endif
 100  if (try.eq.10) then
         print *, 'Sorry.  I cannot seem to solve using the specified'
         print *, ' set of current fractions.  Have a nice day.'
      endif
 110  end


      subroutine setup(a,b,c,d,s,t,delta,delsquare,done)
C
C       This program sets up Newton's method to solve the inverse resistance
C       problem for a network of six resistors and four nodes.
C
C234567-----------------------------------------------------------------|------
C
C      Set up m(1),...,m(6) which are the partial derivatives of delta
```

```
C       with respect to c(1), c(2),...,c(6)
      integer s,t,done
      real*8 a(t,t),b(t),c(s),d(s),delta,delsquare
      real*8 m(6)
      m(1) = c(2)*c(4)+c(2)*c(5)+c(2)*c(6)+c(3)*c(4)+c(3)*c(5)+
     + c(3)*c(6)+c(4)*c(6)+c(5)*c(6)
      m(2) = c(1)*c(4)+c(1)*c(5)+c(1)*c(6)+c(3)*c(4)+c(3)*c(5)+
     + c(3)*c(6)+c(4)*c(5)+c(5)*c(6)
      m(3) = c(1)*c(4)+c(1)*c(5)+c(1)*c(6)+c(2)*c(4)+c(2)*c(5)+
     + c(2)*c(6)+c(4)*c(5)+c(4)*c(6)
      m(4) = c(1)*c(2)+c(1)*c(3)+c(1)*c(6)+c(3)*c(2)+c(2)*c(5)+
     + c(3)*c(6)+c(3)*c(5)+c(5)*c(6)
      m(5) = c(6)*c(4)+c(2)*c(6)+c(1)*c(6)+c(3)*c(4)+c(2)*c(4)+
     + c(3)*c(2)+c(1)*c(3)+c(1)*c(2)
      m(6) = c(1)*c(4)+c(1)*c(5)+c(1)*c(2)+c(3)*c(1)+c(2)*c(5)+
     + c(3)*c(2)+c(4)*c(5)+c(3)*c(4)
C       Set up right-hand side vector
C
      do 12 k=1,5
        b(k) = (c(k)*m(k))/delta - d(k)
 12   continue
      b(6) = c(1)*c(2)*c(3)*c(4)*c(5)*c(6) - 1
      if (abs(b(1)).LE.(.001).and.abs(b(2)).LE.(.001).and.
     + abs(b(3)).LE.(.001).and.abs(b(4)).LE.(.001).and.
     + abs(b(5)).LE.(.001).and.abs(b(6)).LE.(.001)) then
         done=1
         return
      endif
```

```
C       otherwise set up matrix equation and perform Newton's method
      do 15 k = 1,6
         b(k) = -1*delta*b(k)
 15      continue
      a(1,1) = m(1) - c(1)*m(1)**2/delta
      a(1,2) = (c(1)*c(4)+c(1)*c(5)+c(1)*c(6)) - c(1)*m(1)*m(2)/delta
      a(1,3) = (c(1)*c(4)+c(1)*c(5)+c(1)*c(6)) - c(1)*m(1)*m(3)/delta
      a(1,4) = (c(1)*c(2)+c(1)*c(3)+c(1)*c(6)) - c(1)*m(1)*m(4)/delta
      a(1,5) = (c(1)*c(2)+c(1)*c(3)+c(1)*c(6)) - c(1)*m(1)*m(5)/delta
      a(1,6) = (c(1)*c(4)+c(1)*c(5)+c(1)*c(2)+c(1)*c(3))
     +  - c(1)*m(1)*m(6)/delta
      a(2,1) = (c(2)*c(4)+c(2)*c(5)+c(2)*c(6)) - c(2)*m(2)*m(1)/delta
      a(2,2) = m(2) - c(2)*m(2)**2/delta
      a(2,3) = (c(2)*c(4)+c(2)*c(5)+c(2)*c(6)) - c(2)*m(2)*m(3)/delta
      a(2,4) = (c(2)*c(1)+c(2)*c(5)+c(2)*c(3)) - c(2)*m(2)*m(4)/delta
      a(2,5) = (c(2)*c(4)+c(2)*c(3)+c(2)*c(6)+c(2)*c(1))
     +  - c(2)*m(2)*m(5)/delta
      a(2,6) = (c(2)*c(1)+c(2)*c(5)+c(2)*c(3)) - c(2)*m(2)*m(6)/delta
      a(3,1) = (c(3)*c(4)+c(3)*c(5)+c(3)*c(6)) - c(3)*m(3)*m(1)/delta
      a(3,2) = (c(3)*c(4)+c(3)*c(5)+c(3)*c(6)) - c(3)*m(3)*m(2)/delta
      a(3,3) = m(3) - c(3)*m(3)**2/delta
      a(3,4) = (c(3)*c(5)+c(3)*c(1)+c(3)*c(2)+c(3)*c(6))
     +   -c(3)*m(3)*m(4)/delta
      a(3,5) = (c(3)*c(4)+c(3)*c(1)+c(3)*c(2)) - c(3)*m(3)*m(5)/delta
      a(3,6) = (c(3)*c(4)+c(3)*c(1)+c(3)*c(2)) - c(3)*m(3)*m(6)/delta
      a(4,1) = (c(4)*c(2)+c(4)*c(3)+c(4)*c(6)) - c(4)*m(4)*m(1)/delta
      a(4,2) = (c(4)*c(1)+c(4)*c(3)+c(4)*c(5)) - c(4)*m(4)*m(2)/delta
      a(4,3) = (c(4)*c(2)+c(4)*c(1)+c(4)*c(6)+c(4)*c(5))
```

```
+    - c(4)*m(4)*m(3)/delta
 a(4,4) = m(4) - c(4)*m(4)**2/delta
 a(4,5) = (c(4)*c(2)+c(4)*c(3)+c(4)*c(6)) - c(4)*m(4)*m(5)/delta
 a(4,6) = (c(4)*c(1)+c(4)*c(3)+c(4)*c(5)) - c(4)*m(4)*m(6)/delta
 a(5,1) = (c(5)*c(2)+c(5)*c(3)+c(5)*c(6)) - c(5)*m(5)*m(1)/delta
 a(5,2) = (c(5)*c(4)+c(5)*c(3)+c(5)*c(6)+c(5)*c(1))
+     - c(5)*m(5)*m(2)/delta
 a(5,3) = (c(5)*c(2)+c(5)*c(4)+c(5)*c(1)) - c(5)*m(5)*m(3)/delta
 a(5,4) = (c(5)*c(2)+c(5)*c(3)+c(5)*c(6)) - c(5)*m(5)*m(4)/delta
 a(5,5) = m(5) - c(5)*m(5)**2/delta
 a(5,6) = (c(5)*c(2)+c(5)*c(1)+c(5)*c(4)) - c(5)*m(5)*m(6)/delta
 a(6,1) = c(2)*c(3)*c(4)*c(5)*c(6)*delta
 a(6,2) = c(1)*c(3)*c(4)*c(5)*c(6)*delta
 a(6,3) = c(1)*c(2)*c(4)*c(5)*c(6)*delta
 a(6,4) = c(1)*c(2)*c(3)*c(5)*c(6)*delta
 a(6,5) = c(1)*c(2)*c(3)*c(4)*c(6)*delta
 a(6,6) = c(1)*c(2)*c(3)*c(4)*c(5)*delta
 return
  end
```

Bibliography

[1] Bott, R. and Duffin, R.J., "On the Algebra of Networks," *Trans. Amer. Math. Soc.* **74** (1953).

[2] Duffin, R.J., "An Analysis of the Wang Algebra of Networks," *Trans. Amer. Math. Soc.*, (195?).

[3] Dines, K.A. and Lytle, R.J., "Analysis of Electrical Conductivity Imaging," *Geophysics* **46**, No.7, (1981), p.1025-1036.

[4] Gilbert, E.N. and Shepp, L.A., "A Resistance Problem," *SIAM Review* **26** (1984), p.429.

[5] Gilbert, E.N. and Shepp, L.A., "Electrical Tomography," *Open Problems in Communication and Computation,* T. Cover and B. Gopinath, ed., Springer-Verlag, (1987).

[6] Strang, Gilbert, *Introduction to Applied Mathematics*, Wellesley-Cambridge Press, (1986).

[7] Strang, Gilbert, "Inverse Problems and Derivatives of Determinants," Unpublished paper, (1990).

[8] Weinberg, Louis, *Network Analysis and Synthesis*, McGraw-Hill, (1962).